Indoor Localization Without a Prior Map by Trajectory Learning From Crowdsourced Measurements

Jaehyun Yoo, *Member, IEEE*, Karl Henrik Johansson, *Fellow, IEEE*, and Hyoun Jin Kim, *Member, IEEE*

***Abstract*— Accommodation of a situation when a prior map is not available in an indoor localization system is valuable to cost-effective operations by removing a need for map drawing and map updating. This paper suggests a trajectory learning method using crowdsourced measurements in order to support the absence of map. A localization framework based on a particle filter is formalized by machine-learning-based feature extraction and Gaussian process (GP) regression. The feature extraction algorithm reduces dimensionality of sparse measurement vector, and it is applied to detect floor level and designated landmarks. Also, the combination of the feature extraction and the GP regression is used for modeling nonlinear relationship between location and measurement. By this combination, locations of Wi-Fi access points are not required to be known. From the field experimental results, we confirm that the detections of floor level and landmarks are accurate, the learned trajectories are close to the true map, and positioning accuracy is improved thanks to the learning-aided localization.**

***Index Terms*— Crowdsourced measurement, feature extraction, Gaussian process (GP) regression, indoor localization, trajectory learning.**

I. Introduction

INDOOR localization increasingly becomes of interest due to the need for the location information where GPS is not available. Prevalence of wireless access points (APs) built in many buildings and public spaces helps developing received signal strength (RSS)-based indoor localizations using portable devices [1]–[4], with the advantage that installation of additional positioning device, such as ultrawideband [5]–[7], is not necessary.

The main contribution of this paper is to address a situation when a prior map is not available and how to update the change of map without employing a trainer to collect a new database. Given Wi-Fi RSS measurement only, we suggest a trajectory learning method that generates alternative representation of the space instead of geometric map information.

In [8], a trajectory learning algorithm was introduced for Wi-Fi RSS-based indoor localization. During the sampling process, start and end points of a trajectory sample were recognized by extracting Wi-Fi RSS features. After, some representative trajectories are learned by a combination of dynamic time warping and Kalman smoother.

This paper extends the previous work [8] to more realistic localization situation, with the following improvements. First, an extension has been made from a single floor to multiple floors. As well as the landmark detection, the floor detection is done with feature extraction by combination of linear discriminant analysis (LDA) and principal component analysis (PCA). Given a classification model by the LDA and PCA, the k-nearest neighbors (k-NNs) is used for floor detection.

Second, we do not require AP locations. The known AP location is common assumption in RSSi-based indoor localization. However, realizing locations of large amount of AP is difficult and unrealistic in multifloor building. In this paper, the localization framework is formalized by a particle filter whose two major components are likelihood and prior distributions [9]–[11]. The prior distribution is modeled from the learned trajectories via the developed trajectory learning algorithm. The likelihood distribution is modeled by a Gaussian process (GP) regression, which has been popular in recent localization literature [12]–[14] due to its flexibility in representing complex relationships using a small number of parameters [15], [16]. In previous works that used GP as a part of localization architecture, however, AP locations were assumed to be known, which are not suitable for our situation of unknown prior map. In this paper, the combination of the feature extraction and the GP learning plays the role of removing the need for AP locations.

The suggested localization system is evaluated in the multifloor office building. The participants are not given any guideline to carry a smartphone, for example, not to swing the smartphone. From the experimental results, we confirm that the detections of floor level and landmark are accurate and the learned trajectories are close to the true map. The particle filter, which connects each part of the feature extraction, the trajectory learning, and the GP regression, improves the positioning accuracy up to average 1.4 m thanks to the accurate learning performances, in comparison with the results without the learning.

Manuscript received December 4, 2016; revised May 8, 2017; accepted June 13, 2017. This work was supported in part by the Knut and Alice Wallenberg Foundation, in part by the Swedish Research Council, and in part by the Swedish Foundation for Strategic Research, including the SSF-NRF Sweden-Korea Research Program. The Associate Editor coordinating the review process was Dr. Dario Petri. *(Corresponding author: Jaehyun Yoo.)*

J. Yoo and K. H. Johansson are with the ACCESS Linnaeus Center, School of Electrical Engineering, KTH Royal Institute of Technology, 100 44 Stockholm, Sweden (e-mail: jaehyun@kth.se; kallej@kth.se).

H. Jin Kim is with the Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul 08826, South Korea (e-mail: hjinkim@snu.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2017.2729438

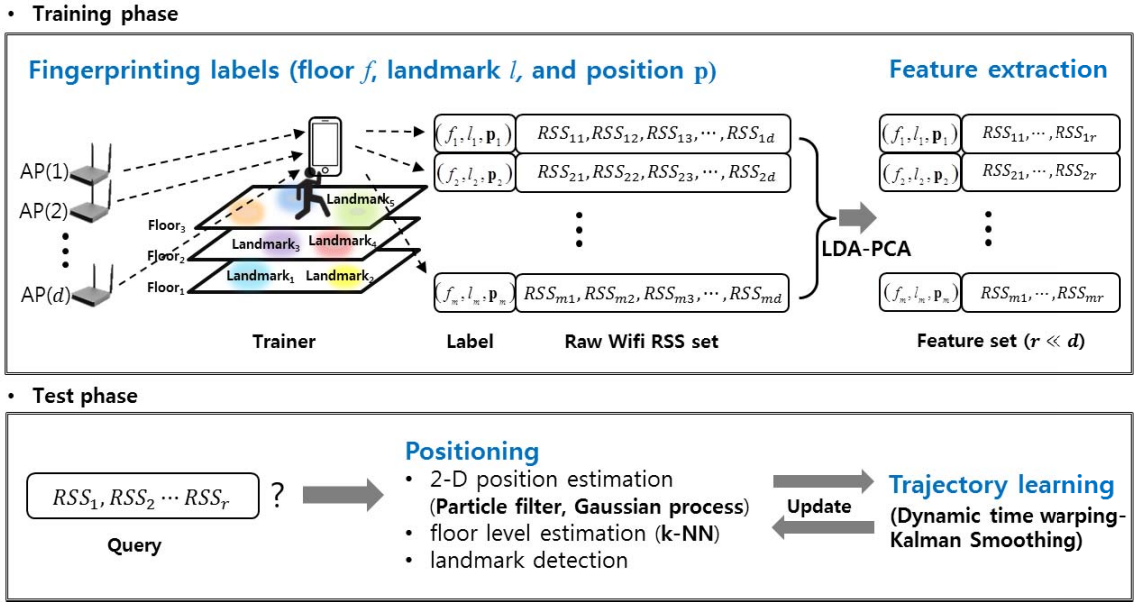


Fig. 1. Indoor localization architecture based on feature extraction, positioning, and trajectory learning.

This paper is organized as follows. Section II describes the related works and Section III formalizes a localization framework based on a particle filter. Section IV introduces the trajectory learning algorithm. In Section V, the combination of GP regression and feature extraction is presented. Section VI presents the experimental results. Sections VII and VIII are devoted to further discussion and concluding remarks.

II. Related Works

This paper considers a Wi-Fi-based indoor localization using smartphone, where true map is unavailable. Fig. 1 shows our indoor localization framework in which each part of feature extraction, trajectory learning, and positioning is connected. This section overviews the related works to feature extraction from Wi-Fi RSS measurements in Section II-A, and localization without a prior map in Section II-B.

A. Feature Extraction From Wi-Fi RSS

Dimensionality of a Wi-Fi RSS vector is decided after recognizing how many different signal identities are included in the training data. High-dimensional Wi-Fi measurement set disturbs realtime operations for smartphone. Besides, many elements in the raw observation set have meaningless (or empty) values, because each AP cannot cover entire indoor area. Therefore, a feature extraction from the raw Wi-Fi RSS set is paramount for localization.

LDA and PCA are both fundamental feature extraction techniques by eliminating worthless elements from original data set. A difference of the LDA and PCA is that PCA uses unlabeled training data (Wi-Fi RSSs only) and the LDA uses labeled training data that are defined as a set of Wi-Fi RSSs and corresponding labels, such as floor level, landmark index, and position, as shown in Fig. 1. Because the LDA tends to cause overfitting problem, the combination with the PCA can improve accuracy by using additional unlabeled data [17]. Moreover, it can address efficiency, because large amount of unlabeled data can be collected without costly effort. Despite a small amount of labeled training data, the combination of the LDA and PCA (or semisupervised discriminant analysis) demonstrates accurate feature extraction performances [18]–[20].

In this paper, a role of the feature extraction algorithm using the semisupervised discriminant analysis is threefold. First, dimensionality of the raw RSS vector is incredibly reduced. Second, the feature extraction with respect to different floors and different landmarks enables us to detect (or classify) floor level and landmarks, to encourage localization performance. The dimensionality reduction and the feature extraction are implemented simultaneously, which is described in Section IV. Third, in Section V, the feature extraction algorithm is combined with a GP regression to overcome the unknown AP locations.

B. Localization Without a Prior Map

Accommodation of the situation without a prior map for indoor localization is valuable, because it can keep privacy of users and service providers, and can remove a need for costly map drawing. The works [4], [21] employ inertial measurement unit sensors for smartphone localization. However, for achieving accurate velocity estimation, a smartphone needs to be kept fixed without rotating. Also, the estimation of heading direction may be biased due to indoor ferrous material and accumulated gyro error. In [22] and [23], RSS propagation models are assumed to be so accurate for localization without map information. However, signal propagation characteristics significantly varies in an indoor area due to the multipath fading problem caused by walls and structure in indoor environments. In [24] and [25], they rely on visual sensors that need to always face forward, which is also a restricted posture, and

thus impractical regarding to crowdsourcing. These restrictions on smartphone-based localization prohibit to apply standard simultaneous localization and mapping techniques [26]–[28], because they should match discrete entities, such as wall and obstacle, by using precise sensors, such as sonar or laser range finders and cameras.

Instead of building geometric map, our approach is to learn a feature-based map by the combination of the feature extraction and the localization scheme. We suggest a localization using crowdsourcing from which we can collect massive amount of qualified training samples without costly efforts. Those obtained samples are trained to find hidden intended trajectories by the developed trajectory learning algorithm. The learned trajectories become an alternative map, and the algorithm is made by dynamic time warping and Kalman smoothing, which originate from the demonstration learning algorithms [29], [30].

III. Problem Formulation

This section formalizes indoor localization using Wi-Fi RSSs that are measurements between a smartphone and wireless APs. The Wi-Fi RSSs collected from indoor environments are nonlinear and noisy by interference of other signals, walls, and obstacles. Furthermore, by assumption of the unknown map information, we do not know real locations of APs. To deal with those problems, machine-learning-based models are presented. In this paper, a particle filter is adopted to formalize the indoor localization framework because of its flexibility to connect the learned models.

Let us define $\mathbf{x}_t$ as location and $\mathbf{y}_t$ as Wi-Fi RSS data set at discrete time step t. Given the history of observations $\mathbf{y}_{1:t}$, the purpose of the particle filter is to estimate posterior probabilistic density function (pdf), $p(\mathbf{x}_t|\mathbf{y}_{1:t})$. Using the particle filter with sequential importance sampling [31], the posterior pdf is obtained by a finite set of weighted particles. Let $\{\mathbf{x}_t^i, w_t^i\}_{i=1}^{N_p}$ be a set of N_p number of particles and weights, where the weights are normalized, such that $\sum_{i=1}^{N_p} w_t^i = 1$. Each $\mathbf{x}_t^i$ represents the hypothetical state of the true state with the corresponding probabilistic value w_t^i. The pdf is given by

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \sum_{i=1}^{N_p} w_t^i \delta\left(\mathbf{x}_t - \mathbf{x}_t^i\right) \tag{1}$$

where $\delta(\cdot)$ is the Dirac delta function. Then, the estimated state can be obtained by

$$\hat{\mathbf{x}}_t = \sum_{i=1}^{N_p} w_t^i \mathbf{x}_t^i. \tag{2}$$

The weights can be obtained in the following:

$$w_t^i = w_{t-1}^i \cdot p\left(\mathbf{y}_t|\mathbf{x}_t^i\right)$$

where the initial weight is given $w_0 = 1/N_p$. The likelihood $p(\mathbf{y}_t|\mathbf{x}_t^i)$ that represents the relationship between location and RSS observation will be specifically formulated in Section V.

On the other hand, the particles are sampled by the following prior probability:

$$\mathbf{x}_t^i \sim p\left(\mathbf{x}_t|\mathbf{x}_{t-1}^i\right). \tag{3}$$

In general localization situation, the prior is modeled by

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) = P_t^{\text{dis}} \cdot P_t^{\text{map}} \tag{4}$$

where

$$P_t^{\text{dis}} = \mathcal{N}(\parallel \mathbf{x}_t - \mathbf{x}_{t-1} \parallel; v\,\Delta t, \sigma_v \Delta t) \tag{5}$$

where $\mathcal{N}$ represents the normal distribution and Δt is the discrete time interval, and

$$P_t^{\text{map}} = \begin{cases} 0 & \text{if a particle crossed a wall} \\ 1 & \text{if a particle did not cross a wall.} \end{cases} \tag{6}$$

The prior functions P_t^{dis} and P_t^{map} keep the localized trajectory smooth and inside reachable area, respectively. Here, we tackle the practical problems about an application of conventional prior functions P^{dis} and P^{map}. The probability P^{dis} needs an accurate velocity measurement v of a user. However, obtaining accurate velocity requires restrictive condition. For example, an attitude for carrying a smartphone has to be kept fixed without rotating, such as foot-mounted pedestrian tracking [4]. This is not practical, because most people swing a smartphone when walking. Instead of P^{dis}, we apply the Hodrick–Prescott filter [32] to suggest the distribution P^{hp} that does not require velocity estimation, given by

$$P_t^{\text{hp}} = \mathcal{N}(\parallel \mathbf{x}_t - \mathbf{x}_{t-2} - 2\mathbf{x}_{t-1} \parallel^2; 0, \sigma_v). \tag{7}$$

P_t^{hp} can obtain a smoothed-curve representation of a time-series trajectory, without velocity information.

Also, because the accuracy of indoor localization considerably relies on the true map P_t^{map}, it is dramatically decreased when the map is unavailable. Our contribution is to generate an alternative function P_t^{cl} that can replace P_t^{map} by learning paths on map. The suggested prior is given by

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) = P_t^{\text{hp}} \cdot P_t^{\text{cl}}. \tag{8}$$

Therefore, the contributions to build the particle filter can be summarized in modeling two distributions of the prior P_t^{cl} (8) and the likelihood $p(\mathbf{y}_t|\mathbf{x}_t)$ (3). Each one will be presented in Sections IV and V, respectively.

IV. Trajectory Learning

This section presents a trajectory learning algorithm using crowdsourced measurements, which will be a part of the prior function P_t^{cl} [readers can see its definition in (28) beforehand].

Given Wi-Fi RSSs as sensory measurements, we suggest a machine learning technique to find the patterns of the Wi-Fi RSSs on some designated landmarks and different floors. First, in Section IV-A, we address how to collect trajectory samples from a crowd. Given Wi-Fi RSSs as sensory measurements, we suggest a machine learning technique to find the patterns of the Wi-Fi RSSs on some designated landmarks and different floors. Second, Section IV-B presents how to learn representative trajectory from the samples, which eventually serves as an alternative representation of the space instead of geometric map information.

A. Sampling Trajectory With Feature Extraction

In a training phase, a trainer should designate landmarks and floors. Then, the trainer collects Wi-FiRSS measurements and assigns labels to them with floor level and landmark index. In order to extract features of those RSSs with respect to the floors and landmarks, this paper suggests a combination of the LDA and PCA.

The LDA and PCA are fundamental dimensionality-reduction algorithms to eliminate uncesessary elements from the original data set, by solving the eigenvalue problem. Let $\mathbf{y} = [y_1, \ldots, y_d]^T \in R^d$ be a vector of raw RSSs obtained from d Wi-Fi APs and $\mathbf{z} \in R^r$ $(1 \leq r \leq d)$ be a transformed vector, where r is the reduced dimensionality. It aims to find a transformation matrix $T \in R^{d \times r}$ such that

$$\mathbf{z} = T^T \mathbf{y} \tag{9}$$

where $\cdot^T$ represents the transpose. The calculation of the transformation matrix involves an optimization problem given by

$$\hat{T} = \text{argmax}_T [\text{tr}(T^T B T (T^T C T)^{-1})] \tag{10}$$

where B and C are the quantities that we want to increase and decrease, respectively, and $\text{tr}(\cdot)$ represents the matrix trace. The matrices B and C can be expressed in a pairwise form [17]. The PCA defines the matrices as

$$B^{\text{PCA}} = \frac{1}{2} \sum_{i,j=1}^{u} W_{ij} (\mathbf{y}_i - \mathbf{y}_j)(\mathbf{y}_i - \mathbf{y}_j)^T \tag{11}$$

$$C^{\text{PCA}} = I \tag{12}$$

with $W_{ij} = 1/u$. The matrix I is the identity matrix and u is the number of unlabeled data.

The LDA defines the matrices as

$$B^{\text{LDA}} = \frac{1}{2} \sum_{i,j=1}^{l} W_{ij}^B (\mathbf{y}_i - \mathbf{y}_j)(\mathbf{y}_i - \mathbf{y}_j)^T \tag{13}$$

$$C^{\text{LDA}} = \frac{1}{2} \sum_{i,j=1}^{l} W_{ij}^C (\mathbf{y}_i - \mathbf{y}_j)(\mathbf{y}_i - \mathbf{y}_j)^T \tag{14}$$

with

$$W_{ij}^B = \begin{cases} \frac{1}{l} - \frac{1}{l_c} & \text{if } w_i = w_j \\ \frac{1}{l} & \text{if } w_i \neq w_j \end{cases} \tag{15}$$

$$W_{ij}^C = \begin{cases} \frac{1}{l_c} & \text{if } w_i = w_j \\ 0 & \text{if } w_i \neq w_j \end{cases} \tag{16}$$

where l is the number of the labeled data and l_c is the number of the labeled samples involved in class c, such that $\sum_c l_c = l$. The labels w_i can be, for instance, a floor level or a landmark index in this paper.

We note that the conventional B^{LDA} and C^{LDA} values [17]–[19] do not involve the unlabeled data. In order to coordinate the unlabeled data to the LDA, we modify them in the following:

$$\bar{B}^{\text{LDA}} = \frac{1}{2} \sum_{i,j=1}^{l+u} \bar{W}_{ij}^B (\mathbf{y}_i - \mathbf{y}_j)(\mathbf{y}_i - \mathbf{y}_j)^T \tag{17}$$

$$\bar{C}^{\text{LDA}} = \frac{1}{2} \sum_{i,j=1}^{l+u} \bar{W}_{ij}^C (\mathbf{y}_i - \mathbf{y}_j)(\mathbf{y}_i - \mathbf{y}_j)^T \tag{18}$$

with

$$\bar{W}_{ij}^B = \begin{cases} \frac{1}{l} - \frac{1}{l_c} & \text{if } w_i = w_j = c \\ \frac{1}{l} & \text{if } w_i \neq w_j \\ \frac{1}{u} & \text{if } w_i = 0 \text{ or } w_j = 0 \end{cases} \tag{19}$$

$$\bar{W}_{ij}^C = \begin{cases} \frac{1}{l_c} & \text{if } w_i = w_j = c \\ 0 & \text{if } w_i \neq w_j \\ 0 & \text{if } w_i = 0 \text{ or } w_j = 0 \end{cases} \tag{20}$$

where the training data set is given by $\{(\mathbf{y}_i, w_i)\}_{i=1}^{l+u}$. If $(\mathbf{y}_i, w_i)$ is a labeled data, $w_i = c$. Otherwise, $w_i = 0$.

The combination is based on weighted summation, given by

$$B = a\bar{B}^{\text{LDA}} + (1-a)B^{\text{PCA}} \tag{21}$$

$$C = a\bar{C}^{\text{LDA}} + (1-a)C^{\text{PCA}} \tag{22}$$

where a is the weight parameter to adjust the balance between the PCA and LDA.

The combination of the PCA and LDA is used for feature extraction. In a training phase, a trainer should designate landmarks and floors. Then, the trainer collects Wi-Fi RSS measurements and assigns labels to them with floor level and landmark index. Using the separate floor and landmark training data, the combination of the LDA and PCA makes each classification model.

In a test phase, given current Wi-Fi RSS observation, it detects which floor and landmark a user is located on.

1) Floor Estimation: Let T^{floor} be the transformation matrix obtained from the training phase when w_i is defined as the floor level and $\mathbf{y}_*$ is the current RSS. Then, $\mathbf{z}_*^{\text{floor}}$ $(=T^{\text{floor}^T}\mathbf{y}_*)$ becomes the transformed current observation. For decision of a test point, the k-NN algorithm is used to assign floor label with the most common among its k-NNs [33].

2) Landmark Detection: Similarly, suppose that T^{land} is the transformation matrix when label w_i is defined as the landmark index, and $\mathbf{z}_*^{\text{land}}$ $(=T^{\text{land}^T}\mathbf{y}_*)$ is the transformed data. We define the distance metric D_c, given by

$$D_c = \left\| \bar{\mu}_c - \mathbf{z}_*^{\text{land}} \right\|, \quad c \in \{\text{landmark}_1, \text{landmark}_2, \ldots\} \tag{23}$$

with

$$\bar{\mu}_c = \frac{1}{l_c} \sum_{i=1:w_i=c}^{l_c} \mathbf{z}_i^{\text{land}} \tag{24}$$

where $\sum_{i=1:w_i=c}^{l_c}$ indicates the summation over $i = 1, \ldots, l_c$, such that $w_i = c$. The metric D_c represents the distance in the feature space between the center of the training datapoints involved to the cth landmark $\bar{\mu}_c$ and the test datapoint $\mathbf{z}_*^{\text{land}}$. The distance metric D_c decreases when a user approaches landmark_c and increases when a user gets far. Thus, we can

detect if a user is located at some interested landmark by thresholding the distance metric. We decide the distance-metric threshold by a 95% confidence interval. Let $\bar{\sigma}_c$ be the standard deviation of samples of cth landmark, and recall the set $\bar{\mu}_c$ and l_c are mean and number of the samples. By the rule of 95% confidence interval, we trigger the detection of the landmark when D_c is smaller than $1.96 \cdot \bar{\sigma}_c / \sqrt{l_c}$.

Before end of this section, we summarize the sampling step. While the user's location is estimated via the particle filter, the floor and the landmark are also detected. Whenever two consecutive landmarks are detected, we recognize them as the start and end points of a user's path. Then, the estimated path between the start and end points becomes one trajectory sample. The trainer gathers a certain amount of the samples and categorizes them according to the label types (the landmark and floor indices). Finally, the trajectory learning algorithm is implemented, which is described in Section IV-B.

B. Learning Trajectory With Kalman Smoother and Dynamic Time Warping

Suppose that M trajectories $\mathbf{X}_j^k$ of length $j = 0, \ldots, N^{(k)} - 1$ and $k = 0, \ldots, M-1$, which have the same start and end points, are given from Section IV-A. Then, there exists a hidden intended trajectory h that is representative of all $\mathbf{X}_j^k$. For example, h can be an average trajectory of $\mathbf{X}_j^k$.

For considering a realistic indoor localization, the samples may have different lengths, because people move at different speeds. Second, some people may generate outlier trajectories although they have the same departure and destination, e.g., detour. Therefore, it is required to detect outliers. We apply a combination of dynamic time warping and Kalman smoother [30] to our trajectory learning approach.

It aims to find one hidden trajectory h_t of length $t = 0, \ldots, O-1$. The size of length O is initially set twice the average length of the trajectories, i.e., $O = 2/M \cdot \sum_{k=1}^{M} N^{(k)}$. The trajectory learning algorithm considers the trajectories $\mathbf{X}_j^k$ as the observations of the one intended trajectory h_t. It is expressed as

$$h_{t+1} = f(h_t) + w_t^h, \quad w^h \sim \mathcal{N}(0, \Sigma^h) \tag{25}$$

$$\mathbf{X}_j^k = h_{\tau_j^k} + w_j^{\mathbf{X}}, \quad w^{\mathbf{X}} \sim \mathcal{N}(0, \Sigma^{\mathbf{X}}) \tag{26}$$

where w^h and $w^{\mathbf{X}}$ are the Gaussian noises whose covariance matrices Σ^z and $\Sigma^{\mathbf{X}}$ are to be estimated. The subscript τ_j^k is time index of h to which the observation $\mathbf{X}_j^k$ is mapped. Fig. 2 is helpful to understand (25) and (26), which shows simplified example results of the trajectory learning. In Fig. 2(a), six different trajectory samples are obtained, and they are represented as $\mathbf{X}_j^k$, with $k = 1, \ldots, 6$. As in (26), these samples are represented by a unique hidden trajectory h_t. Fig. 2(b) shows the resultant trajectory of h_t in (25).

Estimation of the hidden trajectory h_t and the time indices τ_j^k can be done by maximizing the following log likelihood:

$$\max_{\tau, \Sigma^{(\cdot)}} \log p(h, \tau; \Sigma^{(\cdot)}) \tag{27}$$

where $\Sigma^{(\cdot)}$ denotes both Σ^z and $\Sigma^{\mathbf{X}}$. Because it is difficult to optimize the likelihood over $\Sigma^{(\cdot)}$ and τ simultaneously, the previous work [29], [30] reported the optimization algorithm for solving (27). The detailed optimization process is referred to those.

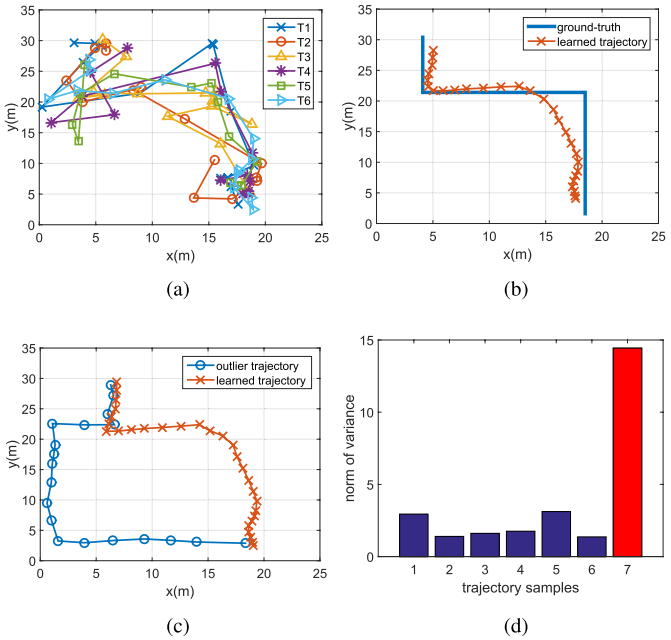


Fig. 2. Experimental example of the trajectory learning algorithm. (a) Trajectory samples. (b) Learned trajectory versus true passage. (c) Added outlier. (d) Outlier detection via variance.

After the learning part, suppose that $\mathbf{h}^{1:n} = \cup_{i=1}^{n} \mathbf{h}^{(i)}$ is a set of n learned trajectories, where each $\mathbf{h}^{(\cdot)}$ has different start and end points. For example, the start and end points of $\mathbf{h}^{(1)}$ are toilet and room, and those of $\mathbf{h}^{(2)}$ are elevator and toilet, respectively. The learned trajectories $\mathbf{h}^{1:n}$ are used for building the probability P_t^{cl}, which follows the Gaussian distribution:

$$P_t^{\mathrm{cl}} \sim \mathcal{N}\left(\mathbf{x}_t; \mu_t^{\mathrm{cl}}, \Sigma_t^{\mathrm{cl}}\right) \tag{28}$$

and

$$\mu_t^{\mathrm{cl}} = \arg_h \min \|\mathbf{h}^{1:n} - \mathbf{x}_t\| \tag{29}$$

where μ_t^{cl} is the closest position among $\mathbf{h}^{1:n}$ to the sample $\mathbf{x}_t$. Also, the variance Σ_t^{cl} is defined as the estimated covariance Σ^{h*} that is obtained from the Kalman smoother.

Fig. 2 shows the experimental results of the trajectory learning when six different people move on the same passage. In Fig. 2(a), individual samples are collected from the localization results of the particle filter. We note that before learning trajectories, the particle filter could not use the prior distribution P_t^{cl}. Thus, the location results, as the trajectory samples in Fig. 2(a), are not smooth and inaccurate. However, their trained result in Fig. 2(b) shows that the learned trajectory is close to the true passage on which the users walked. Fig. 2(c) shows the situation when an outlier is added by the seventh person. We detect the outlier by examining the estimated covariance $\Sigma^{\mathbf{X}}$ that implies the differences between the learned trajectory and the samples. As shown in Fig. 2(d), the outlier has the outstanding norm of variance. We filter an outlier by the 95% confidence interval of trajectory samples.

If a norm of variance of a test sample is outside of the range $(m-1.96\cdot\sigma/\sqrt{n}, m+1.96\cdot\sigma/\sqrt{n})$, where m, σ, and n are the mean, the standard deviation, and the number of the samples, it is treated as an outlier.

V. Gaussian Process Regression With Feature Extraction

This section presents a combination of a feature extraction method and a GP modeling, to build a likelihood relationship between locations and Wi-Fi RSS feature vectors. As the feature extraction algorithm, we adopt the PCA described in Section IV-A, which uses only Wi-Fi RSS (unlabeled data), because we cannot assign labels for Wi-Fi RSSs for the purpose of positioning due to the unknown AP locations. Let $\mathbf{z}_t (= T^{\mathrm{PCA}^T} \cdot \mathbf{y}_t) \in R^r$ be the feature vector to replace the raw observation set $\mathbf{y}_t \in R^d$, such that $r \ll d$. We assume that elements of $\mathbf{z}_t$ are independent. Then, the likelihood is defined as the joint distribution, given by

$$p(\mathbf{z}_t|\mathbf{x}_t) = \prod_{j=1}^{r} p\left(z_t^j|\mathbf{x}_t\right) = \prod_{j=1}^{r} \mathcal{N}\left(\mu_{\mathbf{x}_t}^j, \sigma_{\mathbf{x}_t}^j\right) \tag{30}$$

where z_t^j is the jth scalar component in the vector set $\mathbf{z}_t$ and $\mathcal{N}(\mu_{\mathbf{x}_t}^j, \sigma_{\mathbf{x}_t}^j)$ is Gaussian distribution whose mean $\mu_{\mathbf{x}_t}^j$ and variance $\sigma_{\mathbf{x}_t}^j$ are calculated from a GP regression.

A GP for regression seeks posterior distributions over functions $g(\cdot)$ from training data $\{(\mathbf{x}_i, z_i^j)\}_{i=1}^l$ made by l number of training datapoints drawn from

$$z_i^j = g(\mathbf{x}_i) + \varepsilon \tag{31}$$

where the noise ε follows the Gaussian distribution $\mathcal{N}(0, \sigma_{GP}^2)$. Note that the training output z_i^j in (31) is different to the current feature vector z_t^j. The key idea underlying GP is the requirement that the function values at different datapoints are correlated, where the covariance between two function values $g(\mathbf{x}_p)$ and $g(\mathbf{x}_q)$ depends on the input values $\mathbf{x}_p$ and $\mathbf{x}_q$. This dependence can be specified via the Gaussian kernel function $k(\mathbf{x}_p, \mathbf{x}_q)$, given by

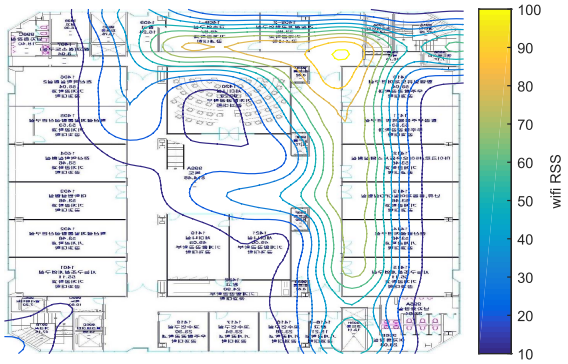
$$k(\mathbf{x}_p, \mathbf{x}_q) = \theta_1 \exp\left(\frac{-\parallel \mathbf{x}_p - \mathbf{x}_q \parallel^2}{2\theta_2}\right) \tag{32}$$

where θ_1 is the signal variance and θ_2 is the length scale that determines how strongly the correlation between datapoints drops off. Both parameters determine the smoothness of the functions estimated by the GP. The parameters can be learned by the conjugate gradient descent method in [34].

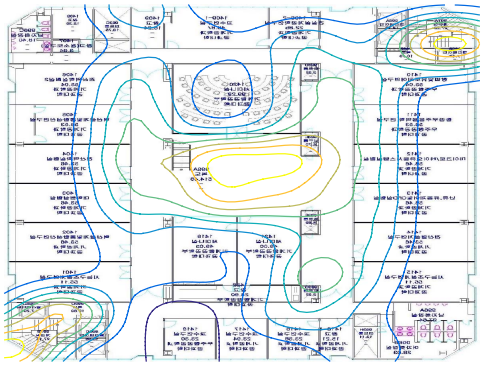
Also, the joint distribution over the training outputs $\mathbf{z}^j = [z_1^j, \ldots, z_l^j]^T$ is a function of the training inputs $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_l]$, with the form

$$\mathbf{z}^j \sim \mathcal{N}\left(0, K(\mathbf{X}, \mathbf{X}) + \sigma_{\mathrm{GP}}^2 I\right) \tag{33}$$

where $K(\mathbf{X}, \mathbf{X})$ is $l \times l$ kernel matrix whose (p, q)th element is $k(\mathbf{x}_p, \mathbf{x}_q)$ in (32). For any set of values $\mathbf{X}$, one can generate the matrix K and then sample a set of corresponding targets $\mathbf{z}^j$.



(a)



(b)

Fig. 3. GP distributions with PCA-driven Wi-Fi RSS feature vectors. (a) GP distribution of the first component in Wi-Fi feature vector. (b) GP distribution of the second component in Wi-Fi feature vector.

After the training, at an interest point $\mathbf{x}_t^*$, the GP estimates the output that takes the Gaussian distribution

$$p\left(g\left(\mathbf{x}_t^*\right)|\mathbf{x}_t^*, \mathbf{X}, \mathbf{z}^j\right) = \mathcal{N}\left(g\left(\mathbf{x}_t^*\right); \mu_{\mathbf{x}_t^*}^j, \sigma_{\mathbf{x}_t^*}^{j^2}\right) \tag{34}$$

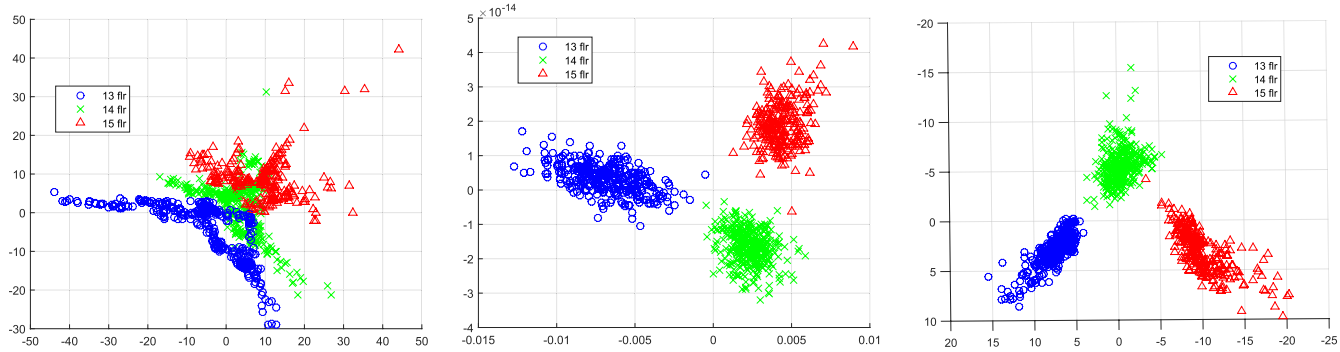
with

$$\mu_{\mathbf{x}_t^*}^j = K\left(\mathbf{X}, \mathbf{x}_t^*\right)^T \left(K(\mathbf{X}, \mathbf{X}) + \sigma_{\mathrm{GP}}^2 I\right)^{-1} \mathbf{z}^j \tag{35}$$

$$\sigma_{\mathbf{x}_t^*}^{j^2} = k\left(\mathbf{x}_t^*, \mathbf{x}_t^*\right) - K\left(\mathbf{X}, \mathbf{x}_t^*\right)^T \left(K(\mathbf{X}, \mathbf{X}) + \sigma_{\mathrm{GP}}^2 I\right)^{-1} K\left(\mathbf{X}, \mathbf{x}_t^*\right) \tag{36}$$

where $K(\mathbf{X}, \mathbf{x}_t^*)$ is the $l \times 1$ vector whose ith element is $k(\mathbf{x}_i, \mathbf{x}_t^*)$. Here, we summarize the estimation process. Initially, we are given training data $\mathbf{X}$ and $\mathbf{z}^j$, and we calculate the kernel matrix $K(\mathbf{X}, \mathbf{X})$. When given a query point $\mathbf{x}_t^*$, GP obtains the kernel vector $K(\mathbf{X}, \mathbf{x}_t^*)$, and then estimates the mean $\mu_{\mathbf{x}_t^*}^j$ and the variance $\sigma_{\mathbf{x}_t^*}^{j^2}$ by (35) and (36). For our localization, we use $\mu_{\mathbf{x}_t^*}^j$ to represent an RSS distribution over space. In the following, we provide the example of the GP estimation for our location by the result of Fig. 3.

Fig. 3 shows the experimental likelihood distributions that indicate $\mu_{\mathbf{x}_t^*}^j$ over one floor, when $\mathbf{x}_t^*$ are defined as 2-D positions on the floor. The original 193-D Wi-Fi RSS ($d = 193$) is reduced to 10-D feature vector ($r = 10$), and 41 position-labeled data are used. Fig. 3(a) shows the



(a) (b) (c)

Fig. 4. Results of (a) PCA, (b) LDA, and (c) combination for floor feature extraction from Wi-Fi RSS training data set with respect to the 13th, 14th, and 15th floors in the experimental building. Training data are collected from 193 APs that are placed over different three floors.

GP distribution of the first components of $\mathbf{z}^1$ in the feature vectors, and Fig. 3(b) shows the distribution of the second components of $\mathbf{z}^2$. We omit the other eight distributions $3 \leq j \leq 10$. From these results, we confirm that the combination of the GP regression and the PCA provides the likelihood distributions without information of the AP locations.

The main difference to general RSS-based localization approach is availability of AP locations. Remind that the likelihood function $p(\mathbf{y}_t|\mathbf{x}_t)$ in (3), where $\mathbf{y}_t \in R^d$ is a set of the RSSs measured from d Wi-Fi APs. If we do not know true locations of the APs, we could not recognize which floor or area is assigned to $p(\mathbf{y}_t|\mathbf{x}_t)$. For example, we cannot draw a GP distribution of a Wi-Fi AP onto the true floor where the AP is located. Therefore, it is useless to apply $p(\mathbf{y}_t|\mathbf{x}_t)$ without the true AP locations. This unrealistic assumption is eliminated in this paper by suggesting the combination of the feature extraction and the GP.

VI. Experimental Results

Indoor experiments are conducted on the 13th, 14th, and 15th floors of the office building 301 at Seoul National University, where the size of the floor is 47 × 36 m^2. Ten different people participate in the experiment, and they are not given any guideline about their posture. We record the user's trajectories by the camera (GoPro Hero3) equipped on the helmet. The experimental videos about the results of the trajectory learning and the integrated localization can be found at `icsl.snu.ac.kr/TrajectoryLearning.avi` and `icsl.snu.ac.kr/WifiIndoorLoc.avi`, respectively.

The used device is Samsung Galaxy S4 whose CPU is 1.6 GHz with android OS and Wi-Fi communication uses the IEEE 802.11 protocol. We program the smartphone application using Java eclipse for obtaining the Wi-Fi signals and measurements, and for communication. The scan function of the Wi-Fi signal in the android platform provides us with information of mac address, AP name, and decibel level of the RSS. On the other hand, the computations for the learning and the localization are performed at the service provider side. When a notebook (i7 CPU 2.60-GHz and 16-GB RAM) is used for the setup with 1324 pieces of datapoints, it took 1.81 s for the PCA–LDA feature extraction and 8.58 s for the trajectory learning.

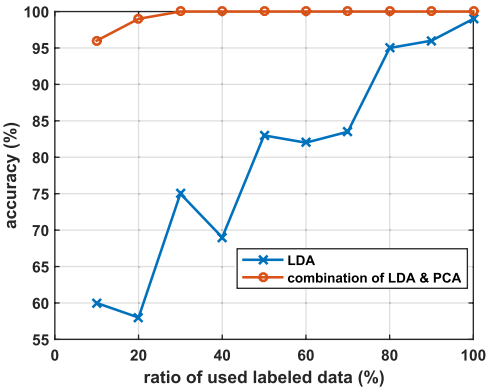


Fig. 5. Accuracy of floor classification using k-NN after: LDA versus combination of LDA and PCA.

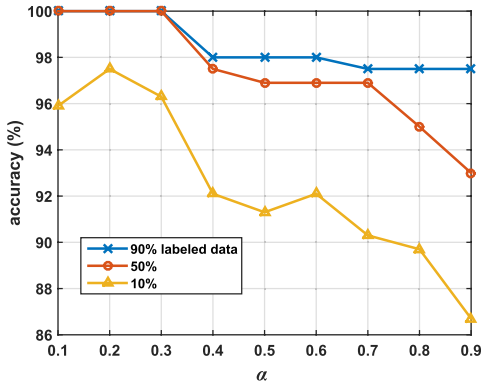


Fig. 6. Accuracy of classification according to the variation of the balancing parameter a in (21).

A total of 193 Wi-Fi APs are placed over the three floors. Due to signal propagation characteristics, some AP signals are found over all floors and some appear on only specific floor. Their variation patterns with respect to floors and location are recognized via the feature extraction algorithm. There are two kinds of training data: labeled data (RSS vectors and corresponding labels) and unlabeled data (RSSs only). The labeled data are also divided into three types. The first

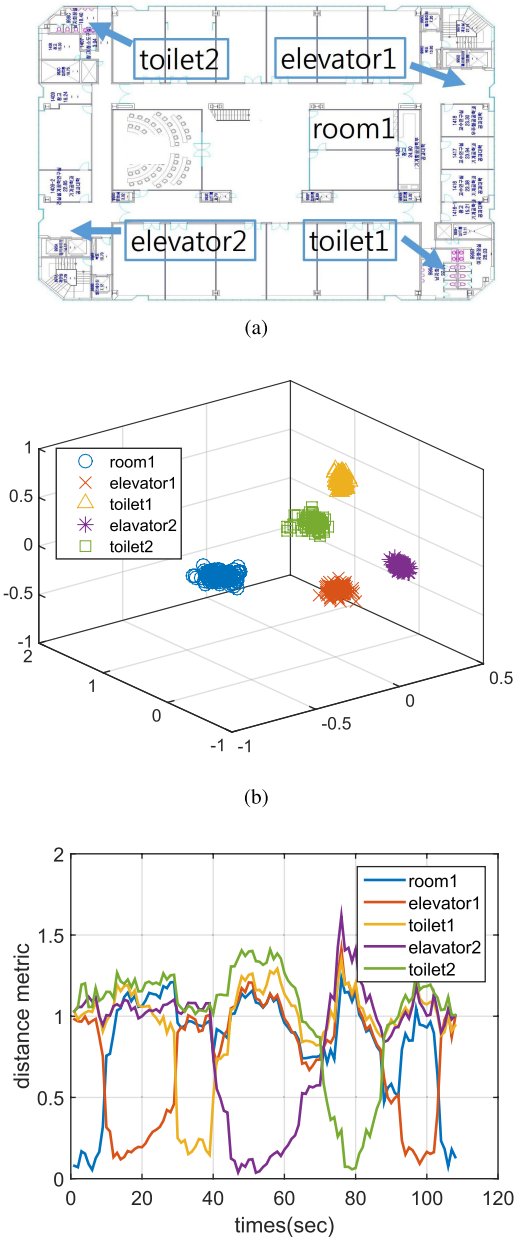


Fig. 7. Landmark detection with (a) designated landmarks on experimental floor, (b) landmark feature extraction from Wi-Fi RSS data set, and (c) variation of the distance metric D_c corresponding to a user's movement: $\text{room}_1 \rightarrow \text{elevator}_1 \rightarrow \text{toilet}_1 \rightarrow \text{elevator}_2 \rightarrow \text{toilet}_2 \rightarrow \text{elevator}_1 \rightarrow \text{room}_1$.

type is the RSS and the corresponding floor level. The second is the RSS and the index of designated landmark. The last is the RSS and the 2-D position.

While the localization, whenever two consecutive landmarks are detected (i.e., start and end points of the trajectory), the estimated trajectory sample is sent to a server. After gathering a certain amount of trajectory samples, we learn new trajectories and update the prior distribution of the particle filter.

A. Floor and Landmark Detections

The dimensionality of Wi-Fi RSS vector is reduced from 193 to 10, and RSSs are scaled from 0 to 1, and the

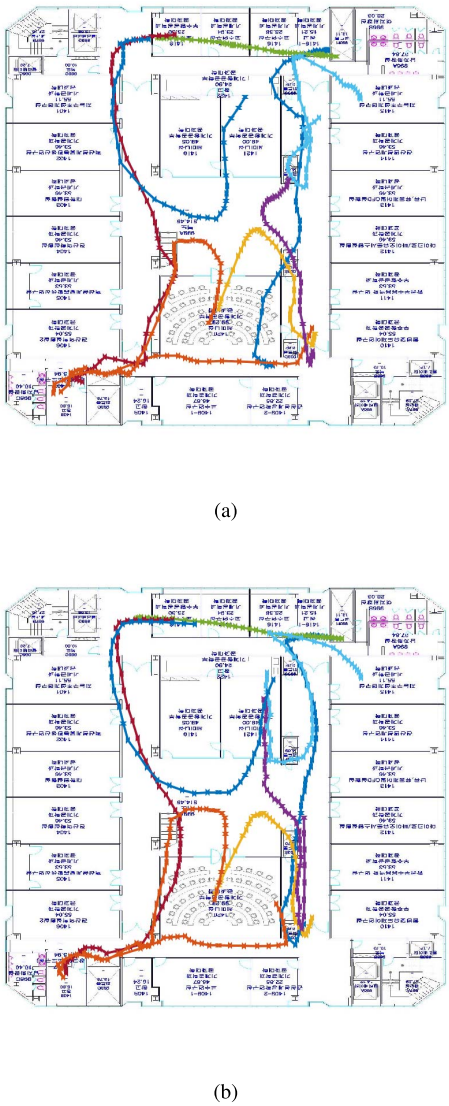
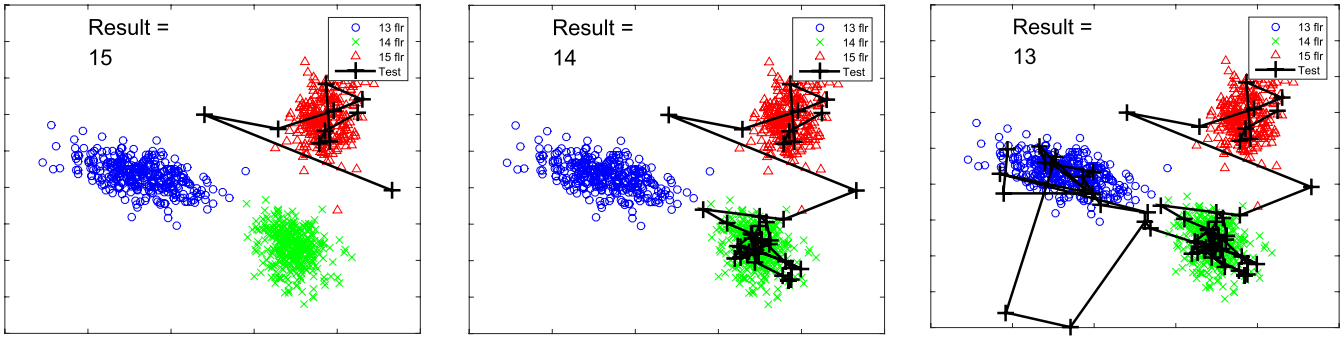


Fig. 8. Learned trajectories with respect to the number of participants. (a) Two participants. (b) Ten participants.

total of 1324 training data points is used. Fig. 4 compares the feature extraction results of the PCA, LDA, and their combination. Fig. 4(a) shows that the PCA is incompetent in the floor classification and the LDA in Fig. 4(b) shows the overfitting of the labeled data by the fact of the different scales of each dimension. In Fig. 4(c), we confirm that the combination algorithm gives more clustered, separated, and scaled result.

Fig. 5 shows the accuracy of the floor classification where k-NN algorithm ($k = 1$) is used to decide the floor level. We vary ratio of the used labeled data from 10% to 100%. For example, when 10% labeled data are used, 90% of the unlabeled data are used. The result of the combination algorithm outperforms the LDA-based classification. A remarkable difference of the accuracy is found when using a small amount of labeled training data, which reflects a situation wherein a trainer can save time and cost for collecting and calibrating training data.

Fig. 6 shows the effect of the balancing parameter a in (21) according to variation of the number of labeled training data.



(a) (b) (c)

Fig. 9. Results of floor detection as well as history of test data on the feature signal space. (a) Time = 30 s. (b) Time = 1.30 min. (c) Time = 3.30 min

(a) (b) (c) (d)

Fig. 10. Results of the position estimation on 14th and 13th floors using the integrated particle filter with the learned map and the GP with feature extraction. (a) Time = 1.5 min on the 14th floor. (b) Time = 2.5 min on the 14th floor. (c) Time = 3.5 min on the 13th floor. (d) Time = 4.5 min on the 13th floor.

We observe that accuracy decreases as value of a increases for three cases. Relatively large a denotes that the semisupervised combination focuses FDA than PCA, which results in inaccurate estimation. In this experiment, the value of a is set 0.2 in accordance with the minimum error shown in Fig. 6.

According to the designated landmarks, such as toilet, room, and elevator as shown in Fig. 7(a), the clear clustering results are shown in Fig. 7(b). Fig. 7(c) evaluates the distance metric with respect to a user's path: $\text{room}_1 \rightarrow \text{elevator}_1 \rightarrow \text{toilet}_1 \rightarrow \text{elevator}_2 \rightarrow \text{toilet}_2 \rightarrow \text{elevator}_1 \rightarrow \text{room}_1$. We can observe the variation of the distance between the designated landmarks and the user location.

B. Trajectory Learning and Positioning

Fig. 8 shows the results of the trajectory learning, where participants walk along one floor randomly. Trajectories made

by the different landmarks are remarked by different colors. Fig. 8(a) and (b) shows the results when samples are collected by two and ten participants, respectively. As more participants join in the learning system, it becomes smoother and closer to the true map.

Fig. 9 shows the history of the test data on the signal space when the user moves from the 15th to 13th floors. Fig. 9(a)–(c) shows the trajectories on the signal space, and we observe that the test points are located onto the each cluster of the (floor) training data sets. The user walks on the stairs when changing floors. The test points on the stairs are marked far from the clusters. The decision of the floor level is done by k-NN ($k = 1$) algorithm.

Fig. 10 shows the positioning result of the particle filter, where we set 50 number of particles. After ten experimental trials, the average error of the developed algorithm is 2.2 m, while an error of the localization without the map learning is 3.6 m. In the worst case, our algorithm has 2.8-m error and the compared one has 5.1 m. In the best case, each has 1.3- and 2.5-m error.

VII. Discussion

This paper devises an indoor localization method by combining the machine learning techniques. In order to remove the need for prior map, the trajectory learning is suggested in which the landmark detection takes an important role of sampling trajectories. Detection of two consecutive landmarks produces one trajectory sample, which can be supported by a landmark selection problem given a fixed number of APs.

In this paper, the clusters of landmarks on Wi-Fi signal space are recognized via the feature extraction algorithm. Therefore, given a fixed number of APs, the landmark selection problem should designate the enough number of landmarks and satisfy certain clustering (or classification) performance. A naive method for landmark selection can be described in the following : 1) initialize N landmark points, whose index set is given $c = \{1, \ldots, N\}$ and 2) find a subset s^* with the maximum cardinality satisfying the clustering (or the classification) performance. This can be represented as $s^* \in \text{argmax}_{s \in c}\{|s| : m(s) < M\}$, where $|s|$ is the cardinality, $m(s)$ is a cost function indicating the clustering performance, and M is a criterion. For instance, $m(s)$ can be defined such that

$$m(s) = \frac{\sum_{i=1}^{|s|} \sum_{j:w_j=i} \|(y_j - \mu_i)\|}{\sum_{i=1}^{|s|} \|(\mu_i - \mu)\|}$$

where y_j is RSS data and w_j is the corresponding landmark labels. And μ and μ_i are the mean of all data and the ith class data, respectively. This function $m(s)$ gauges the degree of clustering capability in that the denominator represents the between-class distance that should be large, while the nominator represents the within-class distance that should be small.

However, the naive approach needs heavy computation in order to test 2^N subsets. On the other hand, the existing landmark-based localization methods [35]–[37] focus on landmark recognition using inertial, magnetic, visual, and GPS sensors, rather than addressing the landmark selection issue using Wi-Fi RSSs. Therefore, more in-depth study for the landmark selection problem needs to be conducted as future work.

VIII. Conclusion

We present an indoor localization with the floor classification, the landmark detection, and the trajectory learning. Wi-Fi RSS is used as sensor observation, but it is not restricted to apply other kinds of sensors. The trajectory learning is developed to update map automatically with crowdsourcing from which we can obtain large amount of qualified training data. For sampling trajectories, we apply the feature extraction algorithm to find patterns of RSSs with respect to different floors and different landmarks. By investigating distances between a test point and the training data points on the feature vector space, we successfully detect the changes of floor and landmark. In order to learn trajectories of unknown map, Kalman smoother and dynamic time warping are applied. As a localization framework, the particle filter that uses the learned map as the prior distribution and the GP modeling as the likelihood distribution is formulated. The experiments are conducted on multiple floors in an office building. Many different people participate in validating experiments and we confirm that the positioning accuracy is improved by the learning-aided localization. The proposed approach can be applied to other types of RSS-based localization, and the performance can be further enhanced by fusing with different types of sensors.

References

[1] P. Pivato, L. Palopoli, and D. Petri, "Accuracy of RSS-based centroid localization algorithms in an indoor environment," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 10, pp. 3451–3460, Oct. 2011.

[2] C. Yang and H.-R. Shao, "WiFi-based indoor positioning," *IEEE Commun. Mag.*, vol. 53, no. 3, pp. 150–157, Mar. 2015.

[3] C.-H. Huang, L.-H. Lee, C. C. Ho, L.-L. Wu, and Z.-H. Lai, "Real-time RFID indoor positioning system based on Kalman-filter drift removal and Heron-bilateration location estimation," *IEEE Trans. Instrum. Meas.*, vol. 64, no. 3, pp. 728–739, Mar. 2015.

[4] A. R. J. Ruiz, F. S. Granja, J. C. P. Honorato, and J. I. G. Rosas, "Accurate pedestrian indoor navigation by tightly coupling foot-mounted IMU and RFID measurements," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 1, pp. 178–189, Jan. 2012.

[5] Y. Zhou, C. L. Law, and F. Chin, "Construction of local anchor map for indoor position measurement system," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 7, pp. 1986–1988, Jul. 2010.

[6] Y. Zhou, C. Law, Y. Guan, and F. Chin, "Indoor elliptical localization based on asynchronous UWB range measurement," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 1, pp. 248–257, Jan. 2011.

[7] A. De Angelis, S. Dwivedi, and P. Händel, "Characterization of a flexible UWB sensor for indoor localization," *IEEE Trans. Instrum. Meas.*, vol. 62, no. 5, pp. 905–913, May 2013.

[8] J. Yoo, H. J. Kim, and K. H. Johansson, "Mapless indoor localization by trajectory learning from a crowd," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Oct. 2016, pp. 1–7.

[9] J. González *et al.*, "Mobile robot localization based on ultra-wide-band ranging: A particle filter approach," *Robot. Auto. Syst.*, vol. 57, no. 5, pp. 496–507, May 2009.

[10] M. Adams, S. Zhang, and L. Xie, "Particle filter based outdoor robot localization using natural features extracted from laser scanners," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, vol. 2. Apr. 2004, pp. 1493–1498.

[11] J. Yoo, W. Kim, and H. J. Kim, "Distributed estimation using online semi-supervised particle filter for mobile sensor networks," *IET Control Theory Appl.*, vol. 9, no. 3, pp. 418–427, 2015.

[12] B. Ferris, D. Fox, and N. Lawrence, "WiFi-SLAM using Gaussian process latent variable models," in *Proc. IJCAI*, vol. 7, 2007, pp. 2480–2485.
[13] F. Seco, C. Plagemann, A. R. Jiménez, and W. Burgard, "Improving RFID-based indoor positioning accuracy using Gaussian processes," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Sep. 2010, pp. 1–8.
[14] A. A. Golovan, A. A. Panyov, V. V. Kosyanchuk, and A. S. Smirnov, "Efficient localization using different mean offset models in Gaussian processes," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Oct. 2014, pp. 365–374.
[15] A. Bermak and S. B. Belhouari, "Bayesian learning using Gaussian process for gas identification," *IEEE Trans. Instrum. Meas.*, vol. 55, no. 3, pp. 787–792, Jun. 2006.
[16] J. H. Yoo, W. Kim, and H. J. Kim, "Event-driven Gaussian process for object localization in wireless sensor networks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 2790–2795.
[17] M. Sugiyama, T. Idé, S. Nakajima, and J. Sese, "Semi-supervised local fisher discriminant analysis for dimensionality reduction," *Mach. Learn.*, vol. 78, nos. 1–2, pp. 35–61, Jan. 2010.
[18] D. Cai, X. He, and J. Han, "Semi-supervised discriminant analysis," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–7.
[19] Y. Zhang and D.-Y. Yeung, "Semi-supervised discriminant analysis using robust path-based similarity," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.
[20] F. Nie, S. Xiang, Y. Jia, and C. Zhang, "Semi-supervised orthogonal discriminant analysis via label propagation," *Pattern Recognit.*, vol. 42, no. 11, pp. 2615–2627, 2009.
[21] J. Á. B. Link, P. Smith, N. Viol, and K. Wehrle, "FootPath: Accurate map-based indoor navigation using smartphones," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Sep. 2011, pp. 1–8.
[22] K. Chintalapudi, A. P. Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," in *Proc. ACM Int. Conf. Mobile Comput. Netw.*, 2010, pp. 173–184.
[23] L. Bruno and P. Robertson, "WiSLAM: Improving FootSLAM with WiFi," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Sep. 2011, pp. 1–10.
[24] M. Werner, M. Kessel, and C. Marouane, "Indoor positioning using smartphone camera," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Sep. 2011, pp. 1–6.
[25] C. Giovannangeli, P. Gaussier, and G. Désilles, "Robust mapless outdoor vision-based navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2006, pp. 3293–3300.
[26] R. Munguía and A. Grau, "Closing loops with a virtual sensor based on monocular SLAM," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 8, pp. 2377–2384, Aug. 2009.
[27] S. Chen and C. Chen, "Probabilistic fuzzy system for uncertain localization and map building of mobile robots," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 6, pp. 1546–1560, Jun. 2012.
[28] T. Yang and V. Aitken, "Evidential mapping for mobile robots with range sensors," *IEEE Trans. Instrum. Meas.*, vol. 55, no. 4, pp. 1422–1429, Aug. 2006.
[29] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proc. ACM Int. Conf. Mach. Learn.*, Jul. 2004, pp. 1–8.
[30] P. Abbeel, A. Coates, and A. Y. Ng, "Autonomous helicopter aerobatics through apprenticeship learning," *Int. J. Robot. Res.*, vol. 29, no. 13, pp. 1608–1639, 2010.
[31] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
[32] M. O. Ravn and H. Uhlig, "On adjusting the hodrick-prescott filter for the frequency of observations," *Rev. Econ. Statist.*, vol. 84, no. 2, pp. 371–376, 2002.
[33] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. IT-13, no. 1, pp. 21–27, Jan. 1967.
[34] C. E. Rasmussen, "Gaussian processes in machine learning," in *Advanced Lectures on Machine Learning*. Berlin, Germany: Springer, 2004, pp. 63–71.
[35] H. Abdelnasser *et al.*, "SemanticSLAM: Using environment landmarks for unsupervised indoor localization," *IEEE Trans. Mobile Comput.*, vol. 15, no. 7, pp. 1770–1782, Jul. 2016.
[36] Y. Bai, W. Jia, H. Zhang, Z.-H. Mao, and M. Sun, "Landmark-based indoor positioning for visually impaired individuals," in *Proc. IEEE Int. Conf. Signal Process.*, Oct. 2014, pp. 668–671.
[37] J. Shang, F. Gu, X. Hu, and A. Kealy, "APFiLoc: An infrastructure-free indoor localization method fusing smartphone inertial sensors, landmarks and map information," *Sensors*, vol. 15, no. 10, pp. 27251–27272, 2015.

Jaehyun Yoo (S'14–M'17) received the B.S. degree in information and control engineering from Kwangwoon University, Seoul, South Korea, in 2010, and the M.S and Ph.D. degrees from the School of Mechanical and Aerospace Engineering, Seoul National University, Seoul, in 2016.

His current research interests include localization, signal processing, automatic control, and also applications of machine learning for robotic systems.

Karl Henrik Johansson (F'13) received the M.Sc. and Ph.D. degrees in electrical engineering from Lund University, Lund, Sweden.

He has held the visiting positions at the University of California, Berkeley, Berkeley, CA, USA, Caltech, Pasadena, CA, USA, Nanyang Technological University, the HKUST Institute of Advanced Studies, and Norwegian University of Science and Technology. He is currently the Director of the Stockholm Strategic Research Area ICT–The Next Generation and the Professor with the School of Electrical Engineering, KTH Royal Institute of Technology, Stockholm, Sweden. His current research interests include networked control systems, cyber-physical systems, and applications in transportation, energy, and automation.

Dr. Johansson is a member of the IEEE Control Systems Society (Board of Governors) and the European Control Association Council. He is a member of the Royal Swedish Academy of Engineering Sciences and an IEEE Distinguished Lecturer. He received several best paper awards and other distinctions, including the ten-year Wallenberg Scholar Grant, the Senior Researcher Position with the Swedish Research Council, the Future Research Leader Award from the Swedish Foundation for Strategic Research, and the triennial Young Author Prize from the International Federation of Automatic Control.

Hyoun Jin Kim (S'98–M'02) received the B.S. degree from the Korea Advanced Institute of Technology, Daejeon, South Korea, in 1995, and the M.S. and Ph.D. degrees in mechanical engineering from the University of California at Berkeley (UC Berkeley), Berkeley, CA, USA, in 1999 and 2001, respectively.

From 2002 to 2004, she was a Post-Doctoral Researcher in electrical engineering and computer science with UC Berkeley. In 2004, she joined the Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul, South Korea, as an Assistant Professor, where she is currently a Professor. Her current research interests include intelligent control of robotic systems and coordination of multirobot systems.