Gaussian Mixture Models for Signal Mapping and Positioning

Matti Raitoharju, Ángel García-Fernández, Roland Hostettler, Robert Piché, and Simo Särkkä

Gaussian Mixture Models for
Signal Mapping and Positioning

M. Raitoharju[a,b,1,*], Á. F. García-Fernández[d,e], R. Hostettler[c,1], R. Piché[b],
S. Särkkä[a,1]

[a] *Aalto University, Espoo, Finland*
[b] *Tampere University, Tampere, Finland*
[c] *Uppsala University, Uppsala, Sweden*
[d] *University of Liverpool, Liverpool, UK*
[e] *Universidad Antonio de Nebrija, Madrid, Spain*

Abstract

Maps of received signal strength (RSS) from a wireless transmitter can be used for positioning or for planning wireless infrastructure. The RSS values measured at a single point are not always the same, but follow some distribution, which vary from point to point. In existing approaches in the literature this variation is neglected or its mapping requires making many measurements at every point, which makes the measurement collection very laborious. We propose to use Gaussian Mixtures (GMs) for modeling joint distributions of the position and the RSS value. The proposed model is more versatile than methods found in the literature as it models the joint distribution of RSS measurements and the location space. This allows us to model the distributions of RSS values in every point of space without making many measurement in every point. In addition, GMs allow us to compute conditional probabilities and posteriors of position in closed form. The proposed models can model any RSS attenuation pattern, which is useful for positioning in multifloor buildings. Our tests with WLAN signals show that positioning with the proposed algorithm provides accurate position estimates. We conclude that the proposed algorithm can provide useful information about distributions of RSS values for different applications.

*Corresponding author. E-mail: matti.raitoharju@aalto.fi
[1] Member of EURASIP

---

1. Introduction

Indoor positioning is often based on signals of existing wireless networks due to the lack of Global Navigation Satellite System (GNSS) signals, high cost of installing and upkeeping dedicated positioning infrastructure, and wide availability of the wireless networks [1]. However, the use of existing wireless infrastructure that is not designed for indoor positioning, such as Wireless Local Area Network (WLAN) or Bluetooth, requires a map of the signal environment [2], a so-called "radio map". Also, consumer WLAN systems provide only received signal strength (RSS) measurements and, for example, range [3], time-of-arrival [4] or direction-of arrival [5] measurements, cannot be used for positioning.

Because walls attenuate and reflect electromagnetic signals and antenna radiation patterns are not directionally uniform, signal strengths cannot be modeled accurately with omnidirectional models, such as path loss in vacuum. Signal attenuation can be modeled using analytic or numerical physics models that take into account the location and materials of the walls [6]. An alternative to physics based models is an empirical model based on a set of measurements of RSS values collected at known locations in the indoor region of interest. These measurements are commonly called Fingerprints (FPs), and there exist many fingerprint positioning methods [7–11].

In this paper, we concentrate on situations where each FP is associated with the location where the measurement was made and the radio map is built first in an offline phase. The map is later used in the online phase for positioning. In the literature, there are also algorithms that map the environment without knowing FP locations [12–15], doing the mapping and positioning in one phase.

One approach to estimate the radio map from FPs is Weighted k Nearest Neighbors (WkNN). Here the position is computed by comparing RSS values with FPs in the database; the estimated position is a weighted average of lo-

2

cations of the k FPs with the most similar RSS values among received Access Points (APs) [16]. The learning phase, thus, consists of collecting the FPs. In the positioning phase, the received RSS values are compared to FPs in the database. In this phase, there are several variations of weighting the k nearest points. The weighting may be based on the difference of RSS values [16] or the rank of the RSSs [17]. The data may also be preprocessed to improve the positioning performance in a varying signal environment [18]. WkNN is commonly used due to its simplicity, but it has two relevant shortcomings: its accuracy can be improved, it does not provide error boundaries for its estimates, and the database size and computational cost increases as more FPs are collected.

Kernel methods and histogram methods [16] extend the WkNN by modeling the distribution of RSS values at calibration points. In the learning phase, multiple measurements are made at each mapped location and the distribution is modeled using a histogram or a kernel method. This increases the required work in the mapping phase significantly and gives only small improvement in the database size or accuracy.

In [19], missing measurement values in FPs are filled and then principal component analysis is applied for dimensionality reduction. Then a Gaussian Mixture Model (GMM) is fitted to model the reduced dimension FPs. Compared to WkNN the computational complexity of the positioning does not increase as the number of samples increase.

In [20], a GMM is proposed to be used to model a the distribution of RSS values inside cells. Compared to the kernel and historgam methods this method allows to track the dependency of RSS values from different APs. In [21] a GMM is used to model the RSS values in the mapped area. The GMM is not used as probabilistic model, but rather as a measurement model with an added Gaussian noise.

Path loss (PL) models are based on a functional relation between signal strength and the distance or path to the receiver [22, 23]. Isotropy may be assumed in open space with omnidirectional antennas. But in built environments, the walls and other structures attenuate and reflect the signals. In particular,

3

the attenuation caused by concrete structures is larger than attenuation from light indoor walls [24]. This anisotropy causes problems if a PL model is to be used in multifloor buildings.

In proximity method the measurement is hard thresholded to provide information whether the measurement was close to an AP [25]. Similarly coverage area methods consider only area where an AP is received with strong enough RSS [26].

Gaussian Process (GP) were used to model radio maps in [27–29]. Their approach has the advantage that it can model RSS patterns of arbitrary distribution in space. However, they need to assume Gaussian noise and the variance of the noise does not depend on the measured RSS values.

Among the above presented methods, only kernel, histogram and GMM methods in [19, 20] are able to model non-Gaussian variations of RSS values, which is useful for quantifying measurement uncertainty in a probabilistic positioning algorithm. However, these methods can only model this characteristic at the points or cells where multiple measurements where collected.

In this paper, we propose a model that fills the above-described gap in the literature. The model uses a Gaussian Mixture (GM) for mapping RSS values in buildings and the map is used for indoor positioning. Specifically we propose a GMM for the joint distribution of RSS values and locations. The main benefit, compared to other methods in literature, is that it models the distribution of the RSS values for each AP in the mapped area without the need of making multiple measurements at each FP location. The proposed model allows us to compute location estimates and RSS distributions in closed form, which allows to evaluate models without approximations.

Compared to GMM based algorithm in [20], the proposed algorithm has the benefit that there is no need for collecting samples in cells as the position dimension is considered as a continuum. Similarly [19] treats the measurement points as separate points without using spatial information. Furthermore, the algorithm in [21] uses the GMM only as a measurement model.

Compared to the Gaussian process methods in [27–29], the proposed model and GP models can model arbitrary signal attenuation patterns inside buildings, but the proposed model can also model arbitrary distributions of RSS values measured at any location close to the area where measurements were made, while GP models rely on the Gaussian distribution.

One important aspect of the model is that it can be used to estimate the distribution of the attenuation and of RSS values in different parts of the building. This is valuable information also for other applications than indoor positioning; for example, RSS maps can help radio infrastructure planners to identify areas that have poor data transfer rates.

The rest of this paper is organized as follows. Section 2 gives the theoretical background of our approach. Section 3 presents the proposed model for building an RSS map. The filtering algorithm based on the proposed model is presented in Section 4. Section 5 presents examples of applying the proposed for WLAN signal mapping and positioning in a four story building. Section 6 concludes the paper.

2. Problem formulation

A Gaussian mixture is a probability density function (pdf) of form

$$p(x) = \sum_{i=1}^{k} w_i p_N(x|\mu_i, P_i), \tag{1}$$

where k is the number of components, $p_N(x|\mu_i, P_i)$ is the pdf of a multivariate normal distribution with mean μ_i and covariance P_i and weights w_i are positive and sum to 1. A Gaussian mixture can approximate any pdf as accurately as desired by increasing the number of components [30].

GMMs can be used in various ways for position estimation. They can be used in Bayesian filters [30]. The generation of components can be automated based on measurements [31] or a generalized form can be used to approximate range measurements [32].

Our objective is to infer the user state $x_t \in \mathbb{R}^n$ at time step t; the state contains horizontal position and velocity, and floor. The information available is the estimate of the user's previous position, the RSS measurements, and RSS maps. For inferring the user's position, we use Bayesian estimation, that is, the posterior is computed from the predicted state $p(x_t \mid x_{t-1})$, which is assumed Markovian, and measurement likelihood $p(y_t, \Omega_t \mid x_t)$, where Ω_t is a vector of Boolean variables describing whether an AP was received, and y_t is a vector of RSS measurements at time index t. Ω_t has value true (T) for the APs that where received and false (F) that were not received. The user state posterior can be written using Bayes' rule:

$$p(x_t \mid \Omega_{1:t}, y_{1:t}) = \frac{p(y_t, \Omega_t \mid x_t)p(x_t \mid \Omega_{1:t-1}, y_{1:t-1})}{\int p(y_t, \Omega_t \mid x_t)p(x_t \mid \Omega_{1:t-1}, y_{1:t-1}) \, \mathrm{d}x_t}, \tag{2}$$

where $p(x_t \mid \Omega_{1:t-1}, y_{1:t-1})$ is the prior distribution and subscript $1 : t-1$ refers to time steps from 1 to $t-1$. The prior distribution can be obtained from the previous posterior using a state transition model $p(x_t \mid x_{t-1})$ as

$$\begin{aligned} p(x_t \mid &\Omega_{1:t-1}, y_{1:t-1}) \\ &= \int p(x_t \mid x_{t-1})p(x_{t-1} \mid \Omega_{1:t-1}, y_{1:t-1}) \, \mathrm{d}x_{t-1}. \end{aligned} \tag{3}$$

This with (2) is the Bayes filter.

If the measurements are conditionally independent, Bayes' rule (2) can be written in the form

$$p(x_t \mid \Omega_{1:t}, y_{1:t}) \propto p(x_t \mid y_{1:t-1}, \Omega_{1:t-1}) \prod_{i=1}^{m} p(y_t^{(i)}, \Omega_t^{(i)} \mid x_t), \tag{4}$$

where $\Omega_t^{(i)}$ indicates whether the ith AP was received, $y_t^{(i)}$ refers to its RSS value and $\propto$ stands for proportionality. Thus the prior can be updated by multiplying it with the product of likelihoods $p(y_t^{(i)}, \Omega_t^{(i)} \mid x_t)$. In reality there can be probabilistic dependence between measurements, but the assumption of independence is made to reduce the modeling effort and computational complexity, and this assumption is commonly made in the literature, for example, in [27, 28, 32, 33].

When an AP is not received ($\Omega_t^{(i)} = \mathrm{F}$) then the corresponding measured RSS is not defined. The absence of sensor readings $\Omega_t^{(i)} = \mathrm{F}$ is so called nega-

tive information and can be used to improve the location estimate [34]; however, we will concentrate on positioning with APs that are received, for which $\Omega = \mathrm{T}$. This is a standard assumption used in the literature [27, 28, 32, 33] with computational benefits, as we do not have to process all APs. Under this assumption, the posterior is

$$p(x_t \mid \Omega_{1:t}, y_{1:t}) \propto p(x_t \mid y_{1:t-1}, \Omega_{1:t-1}) \prod_{\Omega_t^{(i)}=\mathrm{T}} p(y_t^{(i)}, \Omega_t^{(i)} \mid x_t). \tag{5}$$

Goal: The objective of this work is to find a suitable model for the likelihood $p(y_t^{(i)}, \Omega_t^{(i)} \mid x_t)$.

3. Modeling RSS maps with GMs

In this section, we explain how to model RSS maps with GMs. In Section 3.1, we introduce the RSS model. In Section 3.2, we indicate how we can find the parameters for the GMMs.

3.1. State update with RSS model

In order to find a model of the likelihood, we use a fingerprinting approach to train the model. We will model the likelihood indirectly by first modeling the joint distribution

$$p(a, y^{(i)} \mid \Omega^{(i)} = \mathrm{T}) \tag{6}$$

of fingerprinting location a, which is in the same space as the position elements of the state x_t ($\mathbb{R}^2$ for positioning in one floor or $\mathbb{R}^3$ for multifloor positioning), and RSS measurements $y^{(i)}$.

To model the joint distribution (6) of fingerprinting location, including floor number when necessary, and the RSS value, we propose to use GMMs, which can be written as

$$p(a, y^{(i)} \mid \Omega^{(i)} = \mathrm{T}) = \sum_{j=1}^{k} w_j p_N \left(z^{(i)} \mid \mu_j^{(i)}, P_j^{(i)} \right), \tag{7}$$

where $z^{(i)} = \left[\begin{smallmatrix} a \\ y^{(i)} \end{smallmatrix} \right]$, $w_j^{(i)}$ is a component weight, $\mu_j^{(i)}$ is a component mean and $P_j^{(i)}$ is component covariance. By modeling the joint distribution of locations

7

and RSS values with a GMM we do not restrict the shape of the location or RSS distribution. GMM (7) is fitted to the collected fingerprint data. We use notation $\hat{z} = \left[\begin{smallmatrix} \hat{a} \\ \hat{y}^{(i)} \end{smallmatrix}\right]$ for a collected fingerprint to differentiate it from the random variable $z^{(i)} = \left[\begin{smallmatrix} a \\ y^{(i)} \end{smallmatrix}\right]$ that is used to model the distribution of fingerprints. The fitting of the GMM to the collected fingerprints will be explained in Section 3.2.
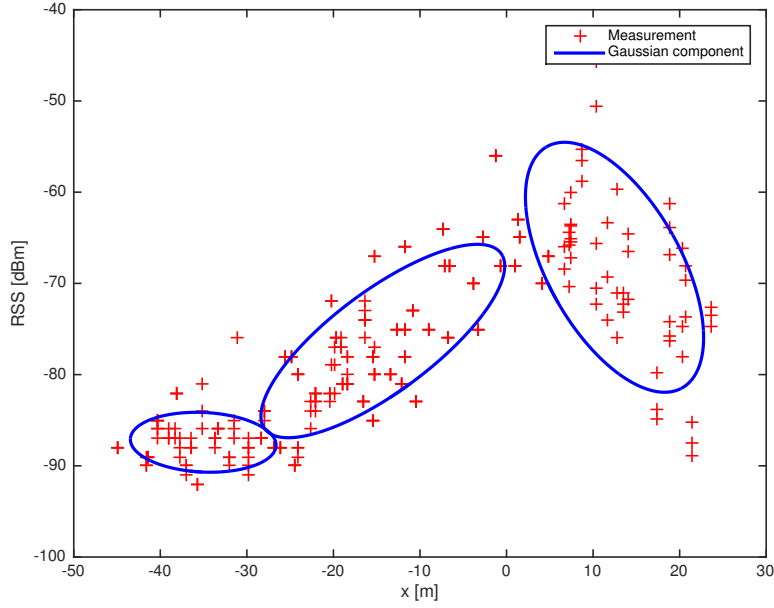

Figure 1: Example of a 3-component GM fitted to one position dimension and RSSs. Each ellipse is an equipotential curve containing 68% of the probability mass of a component.

Example 1. *Figure 1 shows an illustrative example of a 3-component GM that is fitted to data points (red crosses). The points were obtained from real measurements by taking a one-dimensional slice of position dimension and the RSS values. The attenuation is nonlinear and the variation of RSS values depends on x. The proposed model takes this into account by using different variances on different components in RSS dimension. Among the algorithms reviewed in the literature, only the kernel and histogram methods are able to model this kind of variation of RSS values, but only at discrete points where multiple measurements have been taken.*

8

GMs have the property that the product of two GMs is a GM, which can be computed in closed-form (formulas are given in Section 4.2). In this section, we make the necessary assumptions so that the measurement likelihood $p(y_t^{(i)}, \Omega_t^{(i)} = \mathrm{T} \mid x_t)$ in (5) is a GM. Then, we can compute the posterior in closed form using joint GMMs (6) that are constructed using FPs.

We make the following assumptions:

1. Fingerprint locations are uniformly distributed in the region of interest, that is, $p(a) = \mathrm{constant}$.

2. The likelihood for the dynamic state and the fingerprint is the same, that is, $p(y_t^{(i)}, \Omega_t^{(i)} = \mathrm{T} \mid x_t) = p(y_t^{(i)}, \Omega_t^{(i)} = \mathrm{T} \mid a)$ where a is the corresponding position for x_t.

These assumption can be (approximately) met if the FPs are collected uniformly in the area of interest and if the signal environment does not change significantly between FP collection and positioning.

By using Bayes' rule, we can write

$$
\begin{aligned}
p(y^{(i)}, \Omega^{(i)} = \mathrm{T} \mid a) &= \frac{p(a \mid y^{(i)}, \Omega^{(i)} = \mathrm{T}) p(y^{(i)}, \Omega^{(i)} = \mathrm{T})}{p(a)} \\
&= \gamma^{(i)} p(a \mid y^{(i)}, \Omega^{(i)} = \mathrm{T}),
\end{aligned}
\tag{8}
$$

where $\gamma^{(i)} = \frac{p(y^{(i)}, \Omega^{(i)}=\mathrm{T})}{p(a)}$ is a normalization factor that depends only on the AP and measurement value because of our first assumption of uniformly distributed fingerprints. Using the second assumption, the conditional likelihood is

$$
p(a \mid y^{(i)}, \Omega^{(i)} = \mathrm{T}) = p(x_t \mid y^{(i)}, \Omega^{(i)} = \mathrm{T})
\tag{9}
$$

and can be obtained from (6) with closed form formulas that are given in Section 4.1.

Now using (5), (8), and (9), we obtain

$$
\begin{aligned}
p(x_t \mid y_{1:t}, \Omega_{1:t} = T) = {}& p(x_t \mid y_{1:t-1}, \Omega_{1:t-1} = T) \\
& \times \prod_{i:\Omega_t^{(i)}=\mathrm{T}} \gamma^{(i)} p(x_t \mid y_t^{(i)}, \Omega_t^{(i)} = \mathrm{T}),
\end{aligned}
\tag{10}
$$

9

Algorithm 1: Fingerprinting Data preprocessing

---

1 Add random noise sampled from $U(-0.5, 0.5)$ to the RSS and floor index variables. This helps the EM algorithm to avoid singular GM components

2 Split data randomly into:

    Learning set $\hat{z}_1^l, \ldots, \hat{z}_{n^l}^l$

    Validation set $\hat{z}_1^v, \ldots, \hat{z}_{n^v}^v$

3 Compute the covariance P^l of the learning set

4 Compute normalized learning set $\tilde{z}_j^l = \sqrt{P^l}^{-1} z_j^l$, where $\sqrt{P^l}$ is a matrix square root for which $\sqrt{P^l}\sqrt{P^l}^T = P^l$

---

which contains only products of GMs and, thus, the posterior is also a GM. It is not necessary to compute the product of $\gamma^{(i)}$ values as we can normalize the final GM to have $\sum w_j = 1$.

3.2. GMM parameter estimation

In this section, we present an algorithm for computing a GMM for the joint distribution (6) of fingerprint locations and RSS values. First, we must obtain the fingerprints, which contain location and RSS information, in the area of interest. Then we preprocess the fingerprint data and construct k initial clusters using the k-means algorithm. Initial clusters are then refined using the Expectation-Maximization (EM) algorithm. The clustering process is repeated for different values of k to obtain a model with the optimal number of components. Details of this preprocessing are given in the following paragraphs.

In the preprocessing, the data is split into learning and validation sets randomly. The learning set is used to estimate the parameters of the GMM and the validation set is used to decide when to stop the EM iterations. The use of a separate validation set reduces overlearning [35]. For the initial clustering with k-means the data is normalized to get better clustering results. Algorithm 1 presents the preprocessing steps.

The normalized learning data is then clustered using k-means [36]. For initialization of k-means we use the algorithm proposed in [37]. Cluster index c_j refers to the jth cluster and n^{c_j} denotes the number of samples in the jth cluster. In the initialization, the weight of each component is proportional to the number of FPs in it.

The EM algorithm [38] is used to fit a GM (7) to n FPs. We initialize the EM algorithm using the mean and covariance of each cluster produced by the k-means algorithm. The EM algorithm may produce components having singular covariance matrix. To avoid convergence to singular components, we add artificial noise from uniform distribution $U(-0.5, 0.5)$ to the discrete values of RSS and floor in the preprocessing step. Modeling the floor as a continuous variable instead of discrete allows us to use one continuous GMM for an AP instead of a separate GMM for each floor.

The EM algorithm selects k parameters w_j, μ_j, and P_j of the GMM that maximize the joint density of FP locations and RSS values. In logarithmic form, this can be written as

$$\log \prod_{j=1}^{n} \sum_{i=1}^{k} w_i p_N(\hat{z}_j \mid \mu_i, P_i) = \sum_{j=1}^{n} \log \sum_{i=1}^{k} w_i p_N(\hat{z}_j \mid \mu_i, P_i). \qquad (11)$$

At each iteration, the value of (11) increases [38]. We stop the EM iterations when (11) does not improve for the validation dataset. Algorithm 2 shows the pseudocode of the EM algorithm for GM fitting. The number of components k is chosen by fitting a GMM with various values of k and choosing the GMM that produces the best validation likelihood.

Because the k-means algorithm is initialized randomly, the initial clustering and local optimum found by the EM algorithm are random. We build the models so that we start with a single Gaussian ($k = 1$). Then we increment k and build a k-component GMM at most 10 times. We move to the next k if the validation likelihood has not increased from the last iteration and if the validation likelihood is better than with the previous k. When we reach a number of components such that the validation likelihood does not increase

Algorithm 2: Fitting of k component GMM to FPs using the EM algorithm

---

1 Compute initial cluster weights $w_i^- = \frac{n^{c_i}}{n^l}$, where n^{c_i} is the number of FPs in cluster i and n^l is the total number of FPs in learning set

2 Compute initial cluster means $\mu_i^- = \frac{\sum_{j:c_j=i} z_j^l}{n^{c_i}}$, where c_j refers to the index of the cluster to which jth FP belongs

3 Compute initial cluster covariances $P_i^- = \frac{\sum_{j:c_j=i}(z_j - \mu_i^l)(z_j - \mu_i^l)^T}{n^{c_i}}$

4 repeat

5 $\mu_i \leftarrow \mu_i^-, P_i \leftarrow P_i^-, w_i \leftarrow w_i^-$

6   E-step:

7   **for** *each i, j* **do**

8 $\gamma_{ij} \leftarrow \frac{w_i p_N(z_j^l|\mu_i, P_i)}{\sum_{l=1}^{k} w_l p_N(z_j^l|\mu_l, P_l)}$ `// Probability of` j`th sample to`
            `belong to` i`th mixture component`

9   **end**

10   M-step:

11   **for** $i = 1$ *to k* **do**

12 $N_i \leftarrow \sum_{j=1}^{n} \gamma_{ij}$ `// Sum of sample probabilities to belong`
            `to` i`th component`

13 $w_i^- \leftarrow \frac{N_i}{n^l}$ `// Updated weight`

14 $\mu_i^- \leftarrow \frac{1}{N_i} \sum_{j=1}^{n} \gamma_{ij} \hat{z}_j^l$ `// Mean of a component`

15 $P_i^- \leftarrow \frac{1}{N_i} \sum_{c_j=i} \gamma_{ij}(z_j - \mu_i^l)(\hat{z}_j - \mu_i^l)^T$ `// Covariance of a`
            `component`

16   **end**

17 until

$$\sum_{j=1}^{n} \log \sum_{i=1}^{k} w_i p_N(\hat{z}_j^v \mid \mu_i^-, P_i^-) \leq \sum_{j=1}^{n} \log \sum_{i=1}^{k} w_i p_N(\hat{z}_j^v \mid \mu_i, P_i);$$

we choose the GMM having the highest validation likelihood as the resulting model.

4. Filtering using GMs

In this section, we present the Gaussian Mixture Filter (GMF) that will be used for the positioning. In Section 4.1, we show the formulas for computing conditional distributions from GMMs for the GMF and the actual GMF is presented in Section 4.2.

4.1. Computing conditional distribution from a GMM

Conditioning can provide the distribution of RSS when conditioning with a location or the distribution of locations when conditioning on an RSS. To obtain $p(x_t \mid y_t^{(i)}, \Omega_t^{(i)})$ in (10) we need to condition on variable $y_t^{(i)}$. The parameters for the jth component of a conditioned GMM can be derived from the equations used for the GMF in [30] and they are

$$\mu_{j,[x]^{(i)}} \mid y_t^{(i)} = \mu_{j,[x]}^{(i)} + P_{j,[x,y]}^{(i)} \left(P_{j,[y,y]}^{(i)} \right)^{-1} \left(y_t^{(i)} - \mu_{j,[y]}^{(i)} \right) \tag{12a}$$

$$P_{j,[x,x]}^{(i)} \mid y_t^{(i)} = P_{j,[x,x]}^{(i)} - P_{j,[x,y]}^{(i)} \left(P_{j,[y,y]}^{(i)} \right)^{-1} P_{j,[y,x]}^{(i)} \tag{12b}$$

$$w_j^{(i)} \mid y_t^{(i)} \propto w_j^{(i)} p_N \left(y_t^{(i)} \left| \mu_{j,[y]}^{(i)}, \left(P_{j,[y,y]}^{(i)} \right)^{-1} \right. \right), \tag{12c}$$

where $y_t^{(i)}$ refers to measurement value; index set $[y]$ refers to the elements of z that are measurements and $[x]$ refers to the state components that were not used for conditioning.

4.2. GMF for positioning

In this section, we present a GMF that uses the proposed GMMs as a measurement model. We consider that the components of the state vector x_t are ordered so that the first variables describe the location. We also consider that prior density at time 0 is a GM and linear/Gaussian dynamics with a transition density

$$x_t = F x_{t-1} + \varepsilon_Q, \tag{13}$$

where $\varepsilon_Q \sim N(0, Q)$. Then the filtering density is a Gaussian mixture of the form (1).

In the prediction a mixture component is propagated as

$$\mu_{i,t|t-1} = F\mu_{i,t-1|\ t-1} \tag{14a}$$

$$P_{i,t|t-1} = FP_{i,t|t-1}F^T + Q. \tag{14b}$$

In the update, first the measurement GM is conditioned with the corresponding RSS value. Then the parameters of the products of (10) are computed. In computing one product of GMs we compute parameters for each pair of GM components. Updating the ith component of the prior, whose parameters are denoted with superscript (p) to shorten notation, with the jth component of measurement m is done with a Kalman update [30]

$$H = \begin{bmatrix} I & \mathbf{0} \end{bmatrix} \tag{15a}$$

$$S_{i,j}^{(q)} = HP_i^{(m)}H^T + P_j^{(p)} \tag{15b}$$

$$K_{i,j}^{(q)} = P_i^{(p)}H^T \left(S_{i,j}^{(q)}\right)^{-1} \tag{15c}$$

$$\mu_{i,j}^{(q)} = \mu_i^{(p)} + K_{i,j}(\mu_j^{(p)} - H\mu_i^{(m)}) \tag{15d}$$

$$P_{i,j}^{(q)} = P_i - K_{i,j}HP_i^{(p)} \tag{15e}$$

$$w_{i,j}^{(q)} \propto w_i^{(m)}w_j^{(p)}p_N(0; \mu_j^{(p)} - H\mu_i^{(m)}, S_{i,j}^{(q)}), \tag{15f}$$

where superscript (q) refers to the posterior. After updating the components, the weights are normalized so that $\sum_{i,j} w_{i,j} = 1$. This update corresponds to the product in (10).

4.3. Component reduction

Because the number of components of the posterior of two GMs with n and m components has nm components the total number of components increases exponentially with time. To keep the number of components feasible, a component reduction algorithm has to be used. A set of components in a GM can be

collapsed to a smaller set of components in a way that preserves the mean and covariance using the formulas [39]

$$\mu^{(i)} = \sum_j w_j^{(i)} \mu_i^{(i)} \tag{16a}$$

$$P^{(i)} = \sum_j w_j^{(i)} \left(P_i^{(i)} + (\mu^{(i)} - \mu_i^{(i)})(\mu - \mu_i^{(i)})^T \right) \tag{16b}$$

$$w^{(i)} = \sum_j w_j^{(i)}, \tag{16c}$$

where the summations goes over the merged components. In the merging of components some information is always lost. In WLAN positioning there may be, for example, 30 APs available. If each of these APs were modeled with 5 components, the total number of components would be $5^{30} \approx 9 \cdot 10^{20}$. In our implementation, we use the criterion proposed by Runnalls [40] to choose which components should be collapsed.

If the components are reduced after each measurement, the prior and first measurement undergo 30 component reductions, each of which causes approximation errors, and the last measurement will undergo only one reduction. Because the prior contains the information from all previous measurements, we want to have as few component reductions as possible applied to it and also we want to subject all measurements to a similar number of component reductions that cause approximation errors. Thus, we propose to first compute the GM updates between measurement GMFs and last update the prior with the combined measurement GMF.

Figure 2 shows the proposed order of processing applied to GMs from 7 APs. First the APs are conditioned with the RSS measurements using (12b)–(12c). Then these conditional GMs are merged pairwise (GM product ellipses) using (15a)–(15f). Then components are collapsed using Runnalls' algorithm to reduce the number of components. The pairwise merging of APs GMMs is done until there is only one GM left, which is finally used to update the prior GM. Algorithm 3 gives a pseudocode of the proposed GMF including the component reduction.
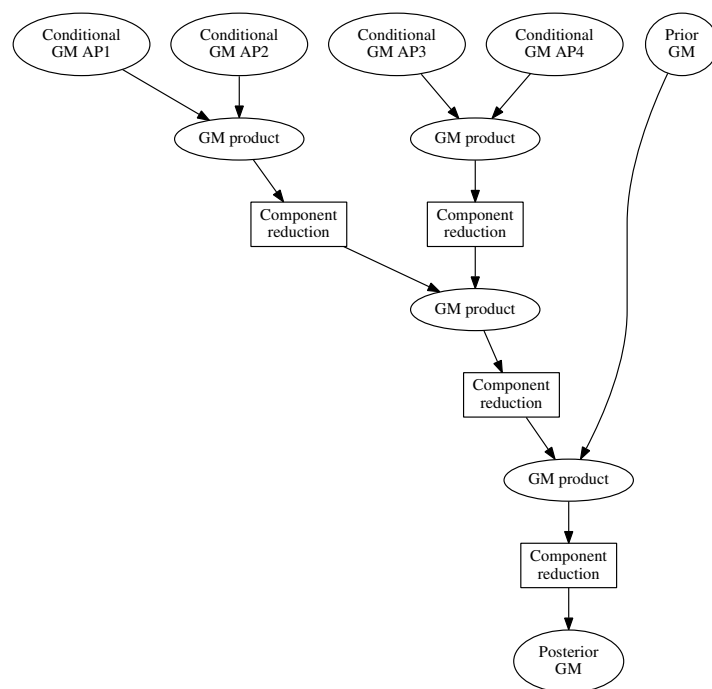
Figure 2: Chart showing how a prior is updated with measurements from 4 APs

---
Algorithm 3: Update step of the GMF for RSS map positioning
---

1 Input:

2   GM parameters of the predicted density of state x_t at the current
    time step t:

3   Prior means $\mu_1^-, \ldots, \mu_k^-$

4   Prior covariances $P_1^-, \ldots, P_k^-$

5   Prior weights $w_1^-, \ldots, w_k^-$

6 k RSS values y_i and corresponding mixture models with parameters
    $w_{i,j}, \mu_{i,j}, P_{i,j}$, where i refers to an AP and j to the jth component
    of ith GM

7 Process measurements:

8   Compute conditional mixture models using (12b)-(12c) for each
    measurement

9   **while** $k{>}1$ **do**

10   **for** $i = 1 : \lfloor \frac{k}{2} \rfloor$ **do**

11   Compute a new ith mixture by updating component $2i-1$
             with component $2i$ using (15a)-(15f)

12   Apply Runnall's algorithm to the new ith mixture

13   **end**

14   **if** k *is odd* **then**

15   Set $\lceil \frac{k}{2} \rceil$th mixture to be kth mixture

16   **end**

17   Set $k \leftarrow \lceil \frac{k}{2} \rceil$

18   **end**

19   Update prior with the combined measurement mixture obtained
    above using (15a)-(15f)

20   Reduce components using Runnalls' algorithm

---

5. Experimental results

In order to test the proposed algorithm we use FPs that were measured from 4 floors of a campus building at Tampere University. Data was collected by a person who walks around the buildings and manually enters the true locations when collecting RSS data. Data was collected with a tablet computer and the number of fingerprints in each floor is: 1530 (Floor 1), 1583 (Floor 2), 333 (Floor 3), and 107 (Floor 4).

5.1. RSS attenuation modeling example

In this example, we aim to illustrate the RSS attenuation model proposed in this paper. We use the data from one AP at the entrance floor of a building. We fit a GMM to the data and then condition the GMM using different RSS values and a floor using (12b) and (12c) with different RSS levels.

Figure 3 shows the contours of conditional probabilities for different RSS levels. Fingerprints that were used for fitting the GM and are close to the conditioning value (within $5\,\mathrm{dBm}$) are shown with dots.

Figure shows how the high probability regions of the conditioned GMM are located where the FP density with similar RSS values is highest, which indicates that the proposed model can capture the distribution of the FPs at different signal levels. We can also note that the attenuation is not isotropic and thus isotropic models cannot model the attenuation accurately.

Now we proceed to illustrate how the variations of RSS measurements from a single AP vary in different parts of a building. As the GMM models the joint distribution, it also models the RSS distribution, which can be obtained for a location by conditioning with its position using (12b) and (12c). In [41], it was observed that the RSS histograms at different locations have different shapes. Figure 4 shows a map with FPs that are color coded to show the RSS values. The four red circles are the locations that are used for the bottom plots, which show RSS density histograms of the measured RSS values within the circles with 5-meter radius and the RSS pdf computed from a GM model. The figure shows

-50dBm

-60dBm

-70dBm

-80dBm

20m

Contours containing 50% and 95% of probability
FP with RSS within 5dBm of nominal

Figure 3: Conditional pdfs of the 2D positions when conditioning with RSS values and floor. For this building we had data from 4 floors. The conditioned contours show how the probability of the conditioned GMM is concentrated on the areas where the learning data was.
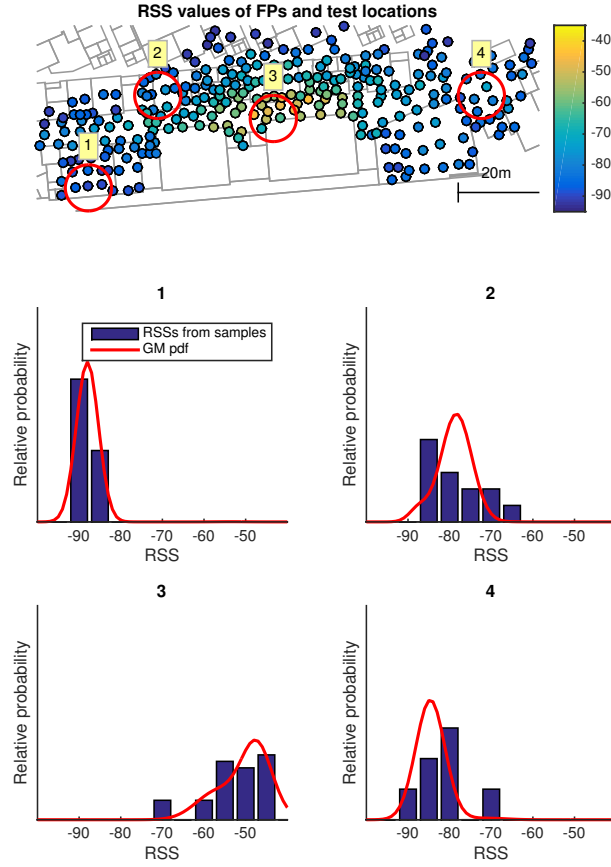
Figure 4: Histograms of RSS values and density of RSS provided by the GMM at 4 different fingerprint locations. These locations are shown in the top figure and numbered from 1 to 4. Histograms are plotted using all FPs that are inside the circles shown in the figure and the densities from GMM are computed in the center of the circles. We can see that the distributions of RSS values are different at different locations and, on the whole, the GMM properly models the RSS measurements at these locations.

how the distribution of the RSS values from a single AP vary in the different parts of the building and how the proposed GM can model this variation.

5.2. Positioning example

Now, we proceed to analyze the positioning accuracy of the proposed GMF with respect to other algorithms in the literature. The test setup and data are the same as used in [7, 42].

Algorithms used in the comparison are two versions of WkNN [16], 1-level Coverage Area (CA) models [43], 2-level CA models [26], PL model [33], and Generalized Gaussian Mixture Filter (GGMF) [32]. For the WkNN, we use 5 nearest neighbors and we implemented the algorithm in two versions. The first version (2D-WkNN) uses known floor and only FPs from that floor. The second version (3D-WkNN) does not use floor information but uses a 3-dimensional position in the FPs. The proposed algorithm and 3D-WkNN are the only algorithms that estimate the floor. The other algorithms use separate maps for each floor and assume known floor level.

The state transition model used is linear

$$x_{t+1} = Fx_t + \varepsilon, \tag{17}$$

where x_t is a 5 dimensional vector whose first component is the position in east direction in local coordinates, second component is the north direction, third dimension is the floor, fourth is velocity in east direction, and fifth is velocity in north direction, F is the state transition matrix, and ε is state transition noise that has zero mean and covariance Q. The state transition matrix is defined as

$$F = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{18}$$

where Δt is the time difference between two measurements in seconds. The state transition noise Q is

$$Q = 0.1 \begin{bmatrix} \frac{\Delta t^3}{3} & 0 & 0 & \frac{\Delta t^2}{2} & 0 \\ 0 & \frac{\Delta t^3}{3} & 0 & 0 & \frac{\Delta t^2}{2} \\ 0 & 0 & \frac{\Delta t}{100} & 0 & 0 \\ \frac{\Delta t^2}{2} & 0 & 0 & \Delta t & 0 \\ 0 & \frac{\Delta t^2}{2} & 0 & 0 & \Delta t \end{bmatrix}. \tag{19}$$

The state was initialized to have correct location mean at first time step with location variance $100^2 I$, which was the same for all algorithms. However, the proposed algorithm could be initialized also using the first set of measurements using processing chart in Figure 2 by leaving the prior GM out of the process.

We tested several values for the parameters of the proposed algorithm and empirically found that assigning 20% of the data to the validation set provided good results. This is in line with the theoretical division rule provided in [44] when we take into account that the number of samples of each AP varies and that the validation set size is data dependent. The maximum number of Gaussian components for a model was set to one tenth of the number of the FPs received from the AP in question.

The test consists of three tracks (pedestrian trajectories) that were walked inside the building. Table 1 shows the errors of routes estimated with different methods in the 3 tracks. The proposed method and 3D-WKNN estimate floor, but the errors are computed only in the horizontal plane. Other methods in the test use only single floor models and are assumed to have access to knowledge of which floor the user is on. Because the radio map generation uses randomness, generated radio maps for the proposed method have variation. We ran the positioning task 100 times and show the mean error for the proposed algorithm. Results show that the proposed algorithm is the most accurate on average. The standard deviation of the accuracy of the proposed algorithm was 0.4 m in the 100 test runs and the proposed algorithm was the most accurate algorithm 85

Table 1: Mean positioning errors of various WLAN positioning methods.

	Error [m]
2D-WKNN	8.2
3D-WKNN	7.9
1-level CA	10.1
2-level CA	7.7
PL	8.3
GGMF	7.8
Proposed	7.2 ± 0.4

times out of 100. Figure 5 shows the Cumulative Distribution Functions (CDFs) of the errors of the proposed algorithm and the 3D-WKNN.

Figure 6 shows the true floors and floor estimates provided by the proposed algorithm and 3D-WKNN (other algorithms do not estimate the floor, but it is given for them). The figure shows how both algorithms provide correct floor estimates most of the times. However, 3D-WKNN has fewer errors. Most of the errors are on Track 2. Figure 7 shows the locations of the erroneous floor estimates. It shows how the errors are clustered around two points. The rightmost is near an elevator and a staircase, where the building has a two floor high open space, and the leftmost is near a staircase. The signals can propagate in staircases more freely than in locations where solid concrete separate the floors and so the signals are more similar near staircases. In the corridors, all the floor estimates are correct.

The proposed algorithm requires 1.16 seconds on average to make a single update. This is suitable for real-time applications in WLAN positioning, as the average processing time is less than making a WLAN scan, which takes a couple of seconds. The algorithms WKNN, CA, PL, and GGMF are faster, taking less than 0.03 s, but the proposed algorithm can tackle real time applications and is more accurate. In addition, the proposed algorithm could be parallelized, which would reduce the computation time.
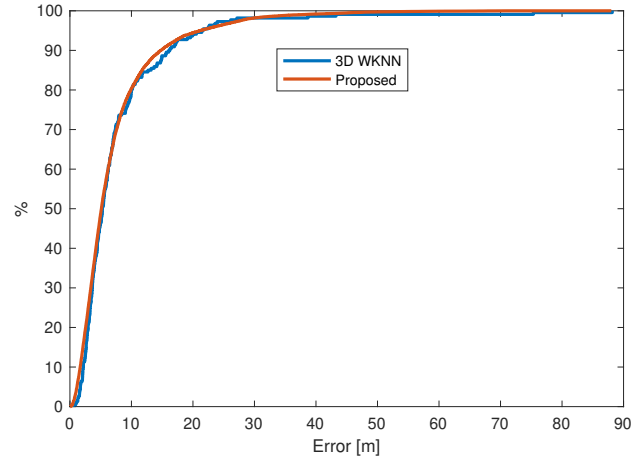
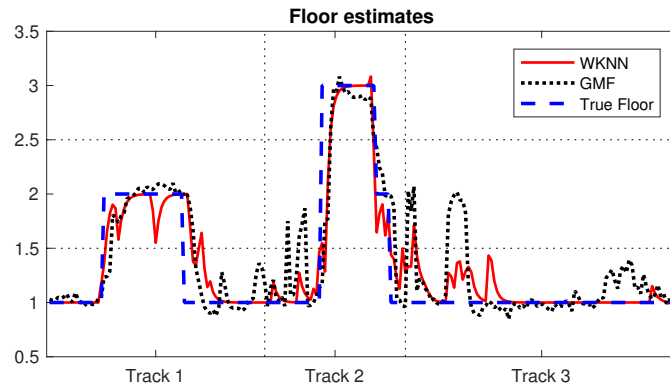Figure 5: CDFs of errors of the proposed algorithm and 3D-WKNN

Figure 6: Estimates of floor computed with 3D-WKNN and the proposed algorithm.

24

Figure 7: Locations of erroneous floor estimates of the proposed algorithm on Track 2.

6. Conclusions and future work

The main novelty this work is a new GMM for WLAN RSS mapping. The new model models the 3D position and RSS values in a single augmented multi-dimensional distribution. The use of a GMM allows to compute the conditional pdfs in closed form. The main findings of the article are:

- Proposed GMM can model the anisotropic attenuation and varying distribution of RSS values inside a building without the requirement to make multitude of measurements at each point.

- In real world positioning tests, the proposed algorithm obtained better accuracy than other WLAN positioning algorithms and was able to estimate the floor accurately.

There are some aspects that should be studied to make the algorithm more versatile. The current algorithm implicitly produces results where the distribution of FPs in the learning phase affects the distribution of positioning. For example, the algorithm estimates that it is more probable to be located in a location where the FP density was high than in a location where FP density is

low. Work is therefore needed to remove this bias and develop a method that copes with non uniform FP sampling density.

There is also room for reducing the computational cost of the proposed algorithm. For instance one could improve the computational efficiency of the learning phase of the algorithm. The proposed algorithm requires processing all FPs in a batch way and does not allow for sequential processing of the FP database, which would be useful for a database that grows with time.

Furthermore, we have so far concentrated only on building maps for a single receiver type even though in reality different devices have different characteristics, for example giving different RSS readings at the same time and place [45]. Building the maps and testing the algorithms for data from different types of devices is another interesting future topic for research.

Acknowledgement

References

[1] Y. Gu, A. Lo, I. Niemegeers, A survey of indoor positioning systems for wireless personal networks, IEEE Communications Surveys Tutorials 11 (1) (2009) 13–32. doi:10.1109/SURV.2009.090103.

[2] M. A. Youssef, A. Agrawala, A. U. Shankar, WLAN location determination via clustering and probability distributions, in: Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. (PerCom 2003)., 2003, pp. 143–150. doi:10.1109/PERCOM.2003.1192736.

[3] L. Martino, J. Míguez, Generalized rejection sampling schemes and applications in signal processing, Signal Processing 90 (11) (2010) 2981–2995.

[4] N. Patwari, A. O. Hero, M. Perkins, N. S. Correal, R. J. O'dea, Relative location estimation in wireless sensor networks, IEEE Transactions on Signal Processing 51 (8) (2003) 2137–2148.

[5] M. Schüssel, Angle of arrival estimation using WiFi and smartphones, in: Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2016, p. 7.

[6] K. El-Kafrawy, M. Youssef, A. El-Keyi, A. Naguib, Propagation modeling for accurate indoor WLAN RSS-based localization, in: 2010 IEEE 72nd Vehicular Technology Conference - Fall, 2010, pp. 1–5. doi:10.1109/VETECF.2010.5594108.

[7] P. Müller, M. Raitoharju, S. Ali-Löytty, L. Wirola, R. Piché, A survey of parametric fingerprint-positioning methods, Gyroscopy and Navigation 7 (2) (2016) 107–127. doi:10.1134/S2075108716020061.

[8] D. Dardari, P. Closas, P. M. Djurić, Indoor tracking: Theory, methods, and technologies, IEEE Transactions on Vehicular Technology 64 (4) (2015) 1263–1278. doi:10.1109/TVT.2015.2403868.

[9] L. Mainetti, L. Patrono, I. Sergi, A survey on indoor positioning systems, in: 2014 22nd International Conference on Software, Telecommunications and Computer Networks (SoftCOM), 2014, pp. 111–120. doi:10.1109/SOFTCOM.2014.7039067.

[10] R. F. Brena, J. P. García-Vázquez, C. E. Galván-Tejada, D. Muñoz-Rodriguez, C. Vargas-Rosales, J. Fangmeyer, Evolution of indoor positioning technologies: A survey, Journal of Sensors 2017. doi:10.1155/2017/2630413.

[11] P. Davidson, R. Piché, A survey of selected indoor positioning methods for smartphones, IEEE Communications Surveys Tutorials 19 (2) (2017) 1347–1370. doi:10.1109/COMST.2016.2637663.

[12] J. Huang, D. Millman, M. Quigley, D. Stavens, S. Thrun, A. Aggarwal, Efficient, generalized indoor WiFi GraphSLAM, in: 2011 IEEE International Conference on Robotics and Automation, 2011, pp. 1038–1043. doi:10.1109/ICRA.2011.5979643.

[13] J. Yoo, K. H. Johansson, H. J. Kim, Indoor localization without a prior map by trajectory learning from crowdsourced measurements, IEEE Transactions on Instrumentation and Measurement 66 (11) (2017) 2825–2835. doi:10.1109/TIM.2017.2729438.

[14] Y. Zhuang, Z. Syed, Y. Li, N. El-Sheimy, Evaluation of two WiFi positioning systems based on autonomous crowdsourcing of handheld devices for indoor navigation, IEEE Transactions on Mobile Computing 15 (8) (2016) 1982–1995. doi:10.1109/TMC.2015.2451641.

[15] B. Ferris, D. Fox, N. D. Lawrence, WiFi-SLAM using Gaussian process latent variable models, in: M. M. Veloso (Ed.), IJCAI, 2007, pp. 2480–2485.

[16] V. Honkavirta, T. Perälä, S. Ali-Löytty, R. Piché, A comparative survey of WLAN location fingerprinting methods, in: Proceedings of the 6th Workshop on Positioning, Navigation and Communication 2009 (WPNC'09), 2009, pp. 243–251. doi:10.1109/WPNC.2009.4907834.

[17] J. Machaj, R. Piché, P. Brida, Rank based fingerprinting algorithm for indoor positioning, in: International Conference on Indoor Positioning and Indoor Navigation (IPIN), 21-23 September 2011, Guimarães, Portugal, 2011, p. 6 p. doi:10.1109/IPIN.2011.6071929.

[18] N. Marques, F. Meneses, A. Moreira, Combining similarity functions and majority rules for multi-building, multi-floor, WiFi positioning, in: 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2012, pp. 1–9. doi:10.1109/IPIN.2012.6418937.

[19] B. A. Akram, A. H. Akbar, K.-H. Kim, CEnsLoc: Infrastructure-less indoor localization methodology using GMM clustering-based classification ensembles, Mobile Information Systems 2018.

[20] Y. Li, S. Williams, B. Moran, A. Kealy, G. Retscher, High-dimensional probabilistic fingerprinting in wireless sensor networks based on a multivariate Gaussian mixture model, Sensors 18 (8) (2018) 2602.

[21] K. Kaji, N. Kawaguchi, Design and implementation of WiFi indoor localization based on Gaussian mixture model and particle filter, in: 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2012, pp. 1–9. doi:10.1109/IPIN.2012.6418943.

[22] T. Rappaport, Wireless Communications: Principles and Practice, 2nd Edition, Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001.

[23] K. Achutegui, J. Míguez, J. Rodas, C. J. Escudero, A multi-model sequential Monte Carlo methodology for indoor tracking: Algorithms and experimental results, Signal Processing 92 (11) (2012) 2594 – 2613. doi:10.1016/j.sigpro.2012.03.017.

[24] D. Micheli, A. Delfini, F. Santoni, F. Volpini, M. Marchetti, Measurement of electromagnetic field attenuation by building walls in the mobile phone and satellite navigation frequency bands, IEEE Antennas and Wireless Propagation Letters 14 (2015) 698–702. doi:10.1109/LAWP.2014.2376811.

[25] Y. Zhao, C. Fritsche, F. Yin, F. Gunnarsson, F. Gustafsson, Sequential Monte Carlo methods and theoretical bounds for proximity report based indoor positioning, IEEE Transactions on Vehicular Technology 67 (6) (2018) 5372–5386.

[26] M. Raitoharju, M. Dashti, S. Ali-Löytty, R. Piché, Positioning with multilevel coverage area models, in: 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN2012), Sydney, Australia, 2012.

[27] B. Ferris, D. Hähnel, D. Fox, Gaussian processes for signal strength-based location estimation, in: In Proc. of Robotics Science and Systems, 2006.

[28] F. Duvallet, A. D. Tews, WiFi position estimation in industrial environments using Gaussian processes, in: 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008, pp. 2216–2221. doi:10.1109/IROS.2008.4650910.

[29] Y. Zhao, C. Liu, L. S. Mihaylova, F. Gunnarsson, Gaussian processes for RSS fingerprints construction in indoor localization, in: 2018 21st International Conference on Information Fusion (FUSION), IEEE, 2018, pp. 1377–1384.

[30] D. Alspach, H. Sorenson, Nonlinear Bayesian estimation using Gaussian sum approximations, IEEE Transactions on Automatic Control 17 (4) (1972) 439–448. doi:10.1109/TAC.1972.1100034.

[31] M. Raitoharju, S. Ali-Löytty, R. Piché, Binomial Gaussian mixture filter, EURASIP Journal on Advances in Signal Processing 2015 (1) (2015) 36. doi:10.1186/s13634-015-0221-2.

[32] P. Müller, H. Wymeersch, R. Piché, UWB positioning with generalized Gaussian mixture filters, IEEE Transactions on Mobile Computing 13 (10) (2014) 2406–2414. doi:10.1109/TMC.2014.2307301.

[33] H. Nurminen, J. Talvitie, S. Ali-Löytty, P. Müller, E.-S. Lohan, R. Piché, M. Renfors, Statistical path loss parameter estimation and positioning using RSS measurements in indoor wireless networks, in: 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN2012), 2012, pp. 1–9. doi:10.1109/IPIN.2012.6418856.

[34] J. Hoffman, M. Spranger, D. Gohring, M. Jungel, Making use of what you don't see: Negative information in Markov localization, in: Intelligent Robots and Systems, 2005. (IROS 2005). 2005

IEEE/RSJ International Conference on, IEEE, 2005, pp. 2947–2952. doi:10.1109/IROS.2005.1545087.

[35] K. J. Coakley, A cross-validation procedure for stopping the EM algorithm and deconvolution of neutron depth profiling spectra, IEEE Transactions on Nuclear Science 38 (1) (1991) 9–15. doi:10.1109/23.64635.

[36] C. M. Bishop, Pattern Recognition and Machine Learning, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[37] D. Arthur, S. Vassilvitskii, k-means++: The advantages of careful seeding, in: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

[38] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society. Series B (Methodological) (1977) 1–38.

[39] Y. Bar-Shalom, R. X. Li, T. Kirubarajan, Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software, John Wiley & Sons, New York, 2001.

[40] A. Runnalls, Kullback-Leibler approach to Gaussian mixture reduction, IEEE Transactions on Aerospace and Electronic Systems 43 (3) (2007) 989–999. doi:10.1109/TAES.2007.4383588.

[41] F. Evennou, F. Marx, Advanced integration of WiFi and inertial navigation systems for indoor mobile positioning, EURASIP Journal on Applied Signal Processing 2006 (2006) 164–164. doi:10.1155/ASP/2006/86706.

[42] P. Müller, M. Raitoharju, R. Piché, A field test of parametric WLAN-fingerprint-positioning methods, in: 17th International Conference on Information Fusion (FUSION), 2014, pp. 1–8.

[43] L. Koski, T. Perälä, R. Piché, Indoor positioning using WLAN coverage area estimates, in: 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Zurich, Switzerland, 2010. doi:10.1109/IPIN.2010.5648284.

[44] I. Guyon, A scaling law for the validation-set training-set size ratio, AT&T Bell Laboratories (1997) 1–11.

[45] Y. Cho, M. Ji, Y. Lee, S. Park, WiFi AP position estimation using contribution from heterogeneous mobile devices, in: Proceedings of the 2012 IEEE/ION Position, Location and Navigation Symposium, 2012, pp. 562–567. doi:10.1109/PLANS.2012.6236928.