A novel Bayesian filtering based algorithm for RSSI-based indoor localization

Faheem Zafari[*], Ioannis Papapanagiotou[⋆⋆], and Thomas J. Hacker[‡]

[*]Electrical and Electronics Engineering, Imperial College London, faheem16@imperial.ac.uk
[⋆⋆]Platform Engineering, Netflix, ipapapa@ncsu.edu
[‡]Computer and Information Technology, Purdue University, tjhacker@purdue.edu

***Abstract*—Indoor localization can provide a number of different services such as location-aware advertisement, indoor navigation and automating different appliances based on the user location. A number of different techniques such as Angle-of-Arrival (AoA), Time-of-Flight (ToF), Time Difference of Arrival (TDoA) and Received Signal Strength Indicator (RSSI) have been used to provide *Location Based Services* (LBS). RSSI is one of the widely used methods as it is cost efficient and easy to implement. However, RSSI's performance is limited by multipath fading and indoor noise. Particle Filter (PF) is an accurate Bayesian Filtering algorithm that can improve the performance of RSSI-based indoor localization. However, PF is not able to satisfy the high accuracy requirement (possibly 10cm) of indoor localization. In this paper, we present *Particle Filter-Extended Kalman Filter* (PFEKF) cascaded algorithm that combines PF and EKF in series to reduce the impact of multipath effects and noise on the RSSI. Our experimental results show that PFEKF improves the localization accuracy by 31.3% and 33.9% in 3D and 2D environments respectively when compared with using only a PF.**

***Index Terms*—Location Based Services, RSSI, Bayesian Filtering, iBeacons**

I. Introduction

The wide-scale use of smartphones, tablets and other wireless devices by multiple users can be leveraged to infer the location of these users and different entities. It can then be used to provide a suite of *Location based services*, services provided to the users based on their location, to the users [1]. However, the presence of different obstacles that can cause multipath fading as well as the existence of noise in indoor environments presents a challenge for accurate indoor localization and tracking. Numerous technologies such as WiFi, Bluetooth, Ultra Wideband (UWB), Acoustic signals, Ultrasound, and Radio Frequency Identification (RFID) and techniques such as Angle-of-Arrival (AoA), Time-of-Flight (ToF), Return Time of Flight (RTOF), Received Signal Strength Indicator (RSSI) and Time Difference of Arrival have been used to obtain an accurate localization system [2]. RSSI is one of the widely used techniques for localization purposes as it is fairly simple to use and does not require complex hardware. However, it is prone to the aforementioned challenges of indoor localization i.e. RSSI is affected by multipath effects and noise [2].

A number of techniques and algorithms have been proposed in the literature to handle the limitations of RSSI and improve its performance in an indoor environment. RSSI fingerprinting is one such technique that relies on a "pre-flight" offline phase site survey to obtain fiducial RSSI values at different points in an indoor environment that are then stored in a database. During the online phase when the user device analyzes the surrounding wireless environment to compute the RSSI values, it compares sensed RSSI values with the fiducial RSSI values from the database. If the sensed RSSI values match with the offline fiducial collected RSSI values, then the user is classified to be located in the position affiliated with the offline RSSI values. However, site surveying to establish fiducial RSSI values and maps is laborious and is not reliable. Especially if the configuration of the indoor environment changes, then there is a concomitant need for another survey as the radio environment changes with the changes in the environment. Bayesian filtering techniques such as Particle Filters (PF), Kalman Filters (KF) and Extended Kalman Filters (EKF) have been used with RSSI based localization and have resulted in improved localization performance. However, the increased demand for high localization accuracy has challenged the research community. Therefore, there is a need for novel algorithms that can improve the localization accuracy of RSSI-based localization systems without incurring significant hardware costs.

In this paper, we present a *Particle Filter-Extended Kalman Filter* (PFEKF) cascaded algorithm that combines PF and EKF in series to reduce the impact of multipath effects and noise on the RSSI. The main contributions of this paper are

- We present a novel cascaded algorithm, *Particle Filter-Extended Kalman Filter*, that improves localization accuracy by 31.3% and 33.9% in 3D and 2D environments respectively when compared with using only a PF.
- We evaluated our algorithm using an iBeacon based prototype that offloads the energy expensive computations to a server to save energy on the user device.
- We also compare our algorithm with our prior work [3] and show that our current algorithm provides encouraging results.

The paper is further structured as: Section II provides a discussion on some of the existing work related to RSSI based indoor localization. Section III describes our algorithm and discusses the indoor tracking model and the Bayesian filters we used. Section IV discusses the experimental setup and provides the results. Section V concludes the paper.

II. Related Work

A number of different indoor localization systems have been proposed over the last couple of years that rely on different wireless technologies and techniques [4], [5], [6], [7], [8]. However, we primarily focus on RSSI based indoor localization systems.

RSSI based indoor localization systems can be classified into either fingerprinting or non-fingerprinting based localization systems. *Fingerprinting* based approaches rely on an offline step in which fiducial RSSI measurements from different reference nodes are stored for use as reference measurements later on. For example, WiFi Access Points, whose position is usually fixed and know to the system. Once the RSSI database is populated, then in the online phase, the sensed RSSI values from reference nodes are compared with the offline fiducial measurements to infer the user position based on the similarity between sensed online and offline measurements. *Horus*, proposed by Youssef et al. [9], is an RSSI (collected from WiFi APs) based localization system that relies on an extensive site survey and fingerprinting. During the offline phase, a radio map of the building is constructed. During the online phase, probabilistic methods are used to obtain an estimate of the user location. Horus attains a median localization accuracy as high as 39 cm in one of the assessed testbeds. Guvenc et al. [10] also used Kalman Filter (KF) to refine the RSSI values of WiFi APs, which resulted in an improved indoor localization accuracy. Their system also relies on fingerprinting to obtain the radio map. The authors show that KF outperforms the moving average method and achieved a median accuracy of 2.5m. *RADAR*, proposed by Bahl et al. [11], is a pioneering work that used RSSI fingerprinting to infer the user's location. While Youssef [9] and Guvenc [10] collected the RSSI values from WiFi APs on a user device to calculate the user location, RADAR collects RSSI values from the user device at the APs to estimate user location i.e. the data packets transmitted by the user devices are collected at AP to estimate the RSSI value and user location. Martin et al. [12] also used an RSSI and fingerprinting mechanism to locate any user. An Android phone application installed on the user device collects the RSSI from different WiFi APs present in the environment. Martin's approach is one of the first that uses the same device for both the offline and online phases. A localization error as high as 1.5m was reported.

Non-fingerprinting based approaches do not require a laborious "pre-flight" site survey. Small scale calibrations required to obtain the path-loss coefficient are carried out by the localization system administrator. The path-loss coefficient is used in the log-normal shadowing model described by Kumar et al. [7] to map the RSSI values into distance. In our prior work [13], we used Particle Filters to improve the performance of iBeacons for indoor localization. Using the RSSI values of the beacon message, we used n-point trilateration in conjunction with the PF. We attained a localization accuracy as high as 0.97 meters. In our prior work [14], [3], we used the RSSI values of iBeacons for estimating user proximity to any *Point of Interest* (PoI). Using moving average and Kalman Filters, iBeacon's proximity detection accuracy was improved by 29% and 32% respectively in comparison with current approach adopted by iBeacon protocol. In our recent work [3], we use a Kalman filter (KF) in cascade with PF to improve the localization accuracy of an iBeacon based indoor localization system by 28.16% and 25.59% in 2D and 3D localization respectively when compared with using only PF. In contrast with other related work previously discussed in this section [10], [9], [11], [12], our PFEKF approach does not require any extensive fingerprinting, resulting in a less complex approach. Furthermore, we achieve a comparable localization accuracy. In contrast with our previous efforts [13], our new PFEKF approach achieves a comparatively higher accuracy. Furthermore, we are also able to track the user device in 3D with a much lower energy cost by offloading the computationally expensive aspect of the localization process to a server. While in our prior work [14], we only dealt with proximity detection, PFEKF can provide user location and easily be extended for proximity detection. PFEKF also outperforms our KFPF algorithm described in our prior work in [3] and improves the localization accuracy by 7.6% and 8% in 3D and 2D environments respectively.

III. Particle Filter-Extended Kalman Filter (PFEKF)

Before we discuss the PFEKF algorithm in detail, we first describe the indoor localization model that we use to track the user, and discuss PF and EKF.

A. Indoor Localization Model

We model the indoor localization problem as posed by Arulampalam et al. [15] and used in our prior work [13], [3]. Since we seek to estimate the user position/state under a set of measurements obtained in a typical noisy indoor environment, Bayesian filtering is an attractive approach for such problems. However, Bayesian filtering requires the following two models.

1) System Model: A system model describes the variation of the state (user position in our case) with time. The system model relates the position vector y_i with the process noise m_i and previous state.
2) Measurement Model: A measurement model relates the noisy measurements (RSSI for PF and the user position for EKF) with the state/position.

We construct the posterior probability density function (pdf) of the state by using all the available information, including the measurements from the reference nodes (iBeacons in our case). The pdf is considered as the complete solution to the state estimation problem, since it contains all the required information. Our problem involves recursively estimating the user state/position as we receive measurements from the sensor. Therefore, we require a recursive filter. Recursive filters consist of the prediction and update stage in which the state is predicted and then updated once the measurements are available. The presence of noise in indoor settings affects the position calculation so the pdf is usually distorted. The obtained measurements in the update state are used to modify the prediction pdf using Bayes theorem.

Mathematically, state y_i at time i is a function of the state at time step $(i-1)$ as well as the process noise m_{i-1} [16] as described in Equation (1):

$$y_i = f_i(y_{i-1}, m_{i-1}) \tag{1}$$

$f_i : \Re^{n_y}$ x $\Re^{n_m} \rightarrow \Re^{n_y}$ is the non-linear function (as indoor localization is a non-linear problem) that relates the previous state y_{i-1} and process noise m_{i-1} with the current state y_i as described by Arulampalam et al. [15]. The sequence $\{m_i, i \in \aleph\}$ represents an independent, identically distributed (i.i.d) process noise sequence. The integers n_y and n_m represent the state and process noise vector dimensions respectively. $\aleph$ represents the set of Natural numbers. The measurement model relates the obtained measurement x_i to the state y and measurement noise n at time i [16] as given in Equation (2):

$$x_i = h_i(y_i, n_i) \tag{2}$$

The mapping function $h_i : \Re^{n_y}$ x $\Re^{n_n} \rightarrow \Re^{n_x}$ can be either linear or non-linear. Both functions f_i, and h_i relies on the laws of motion/physics. The sequence $\{n_i, i \in \aleph\}$ is an i.i.d measurement noise sequence. The integers n_x and n_n represent the measurement and measurement noise vectors dimension respectively.

Recursively calculating the pdf $p(y_i|x_{1:i})$ allows us to continuously calculate the belief in the state y_i at any particular time instance i in the presence of noisy measurements. The initial pdf $p(y_o|x_0)$ is assumed to be equivalent to state vector's prior $p(y_0)$ [15]. We assume that the prior is available. The available information is enough to calculate the pdf $p(y_i|x_{1:i})$ recursively in the prediction and update stages. In the prediction stage if the pdf $p(y_{i-1}|x_{1:i-1})$ is available, we can use Chapman-Kolmogorov equation given in Equation (3) to obtain the prior pdf of the state at any time instance i.

$$p(y_i|x_{1:i-1}) = \int p(y_i|y_{i-1})p(y_{k-1}|x_{1:i-1})dy_{i-1} \tag{3}$$

At any time instance i, we collect the observations x_i from the sensors to update the prior using Bayes rule given in Equation (4) [15]. The denominator in Equation (4) is explained in Equation (5).

$$p(y_i|x_{1:i}) = \frac{p(x_i|y_i)p(y_i|x_{1:i-1})}{p(x_i|x_{i-1})} \tag{4}$$

$$p(x_i|x_{i-1}) = \int p(x_i|y_i)p(y_i|x_{i-1})dy_i \tag{5}$$

The collected measurements x_i in the update stage are then used to update the prior density, resulting in the required current state's posterior density. Recursively updating the system using Equations (3) and (4) result in an optimal Bayesian solution. However analytically, it is not possible to obtain the recursive propagation of posterior probability density as done in Equations (3) and (4). Therefore, a number of different algorithms including PF and EKF are used to obtain a solution. Below we discuss the theory of PF and EKF from localization perspective.

B. Particle Filter

Particle filters is widely used for indoor localization and tracking [17]. The basic idea behind particle filters is that the posterior probability distribution is represented using a set of weighted random samples that are used for computing the estimates [15]. An increase in the number of samples cause the filter to perform optimally. To understand the algorithm in detail, first we offer the following summary of the approach. Let $\left\{y^k_{0:i}, w^k_i\right\}$ be the set of random measures that characterize the posterior pdf $p(y_{0:i}|x_{1:i})$. $\left\{y^k_{0:i}, k = 0, ...N_s\right\}$ is the set of support points whereas the weight are given by $\left\{w^k_i, k = 0, ...N_s\right\}$. $y_{0:i}$ where $\left\{y_j, j = 0, ...i\right\}$ is the set of the states up to i. The weights are normalized using $\sum_{min} w^k_i = 1$. After the normalization, the posterior density at i is approximated, as given by [15], using

$$p(y_{0:i}|x_{1:i}) \approx \sum_{k=1}^{N_s} w^k_i \delta(y_{0:i} - y^k_{0:i}) \tag{6}$$

Equation (6) is the discrete weighted approximation of the true posterior probability distribution $p(y_{0:i}|x_{1:i})$. Importance sampling [18] is used to choose the weights associated with each particle [15]. For importance sampling, assume that $p(y) \propto \pi(y)$ is the probability density from which drawing particles is tedious. However, $\pi(y)$ can be evaluated for the probability density. Let $y^k \sim d(y)$ where $k \in [1, ..., M_s]$ be the samples generated from the proposal $d(.)$ known as importance density. Then the probability density $p(.)$ can be approximated as given by [15]

$$p(y) \approx \sum_{k=1}^{M_s} w^k \delta(y - y^k) \tag{7}$$

where as the normalized weight of the k^{th} particle can be obtained using Equation (8)

$$w^k \propto \frac{\pi(y^k)}{d(y^k)} \tag{8}$$

If the samples $y^k_{0:i}$ are taken from the importance density $d(y^k_{0:i}|x_{1:i})$, then the weights used in Equation (6) are given by

$$w^k_i \propto \frac{p(y^k_{0:i}|x_{1:i})}{d(y^k_{0:i}|x_{1:i})} \tag{9}$$

Due to the sequential nature of the process, at every single iteration, there could be samples that approximate the conditional probability $p(y_{0:i-1}|x_{1:i-1})$, and the goal is to approximate $p(y_{0:i}|x_{1:i})$ conditional probability using new samples. The importance density must be chosen for factorizing such that

$$d(y_{0:i}|x_{1:i}) = p(y_i|y_{0:i-1}, x_{i:i})d(y_{0:i-1}|x_{1:i-1}) \tag{10}$$

Then the samples $y^k_{0:i} \sim d(y_{0:i}|x_{1:i})$ can be obtained by incorporating the existing samples $y^k_{0:i-1} \sim d(y_{0:i-1}|x_{1:i-1})$ into the recently obtained state $y^k_i \sim d(y_i|y_{0:i-1}, x_{1:i})$. The weights must be updated using the update Equation (4) that can be derived by first representing $p(y_{0:i}|x_{1:i})$ in terms of $p(x_i|y_i)$, $p(y_i|y_{i-1})$ and $p(y_{0:i-1}|x_{1:i-1})$ that can be mathematically, as described by Arulampalam et al. [15], given by

$$
\begin{aligned}
p(y_{0:i}|x_{1:i}) &= \frac{p(x_i|y_{0:i}|x_{0:i-1})p(y_{0:i}|x_{1:i-1})}{p(x_i|x_{1:i-1})} \\
&= \frac{p(x_i|y_{0:i}|x_{0:i-1})p(y_{0:i}|y_{0:i-1}|x_{1:i-1})}{p(x_i|x_{1:i-1})} \qquad (11) \\
&\quad \times p(y_{0:i-1}|x_{1:i-1}) \\
&= \frac{p(x_i|y_i)p(y_i|y_{i-1})}{p(x_i|x_{1:i-1})}\, p(y_{0:i-1}|x_{1:i-1}) \qquad (12) \\
&\propto p(x_i|y_i)p(y_i|y_{i-1})p(y_{0:i-1}|x_{1:i-1})
\end{aligned}
$$

The weights computed using Equation (9) can be updated by substituting Equation (10) and (12) into it, so Equation (9) becomes

$$
\begin{aligned}
w_i^k &\propto \frac{p(x_i|y_i^k)p(y_i^k|y_{i-1}^k)p(y_{0:i-1}^k|x_{1:i-1})}{d(y_i^k|y_{1:i-1}^k, x_{1:i})d(y_{0:i-1}^k|x_{1:i-1})} \\
&= w_{i-1}^k \frac{p(x_i|y_i^k))p(y_i^k|y_{i-1}^k)}{d(y_i^k|y_{1:i-1}^k, x_{1:i})}
\end{aligned} \qquad (13)
$$

Also the importance density function would be only dependent on y_{i-1} x_i if $d(y_i|y_{1:i-1}, x_{1:i}) = d(y_i|y_{i-1}, x_i)$. This facilitates when only a filtered estimate of $p(y_i|x_{1:i})$ is needed at each time step. In such cases, only the state y_i^k would need to be stored. The weights are then modified, as described by Arulampalam et al. [15], into

$$
w_i^k \propto w_{i-1}^k \frac{p(x_i|y_i^k))p(y_i^k|y_{i-1}^k)}{d(y_i^k|y_{i-1}^k, x_i)} \qquad (14)
$$

while the filtered posterior probability density becomes

$$
p(y_i|x_{1:i}) \approx \sum_{k=1}^{N_s} w_i^k \delta(y_i - y_i^k) \qquad (15)
$$

The aforementioned algorithm (SIS) is a recursive algorithm in which the weights and support points are recursively propagated with the reception of every single measurement. Particle Filters are optimal for indoor localization since they assume the system to be non-linear and noise to be Non-Gaussian which is a realistic assumption for indoor environments.

C. Extended Kalman Filter

While Kalman filter relies on the assumption that the functions given in Equations (1) and (2) are linear, it is not always the case particularly in an indoor environment. So in such cases we need a local linearization of the equations can approximate the non-linearity condition. This is the core essence of Extended Kalman Filters (EKF) as they locally linearize non-linear functions by taking the Jacobian of the non-linear $f(.)$ and $h(.)$ functions listed in Equations (1) and (2) respectively. Let $\bar{F}$ and $\bar{H}$ are the locally linearized functions obtained through the Jacobian of $f(.)$ and $g(.)$ functions. EKF assumes that the probability $p(y_i|x_{1:i})$ can be approximated using Gaussian as given in Equations (16)-(18) by [15]

$$
p(y_{i-1}|x_{1:i-1}) \approx \mathcal{N}(y_{i-1}; m_{i-1|i-1}, P_{i-1|i-1}) \qquad (16)
$$

$$
p(y_i|x_{1:i-1}) \approx \mathcal{N}(y_i; m_{i|i-1}, P_{i|i-1}) \qquad (17)
$$

$$
p(y_i|x_{1:i}) \approx \mathcal{N}(y_i; m_{i|i}, P_{i|i}) \qquad (18)
$$

where $\mathcal{N}(y; m, P)$ is a Gaussian Probability Density and has arguments state y, mean m and covariance P. Similarly (taken from [15])

$$
m_{i|i-1} = f_i(m_{i-1|i-1}) \qquad (19)
$$

$$
P_{i|i-1} = Q_{i-1} + \bar{F}_i P_{i-1|i} \bar{F}_i^T \qquad (20)
$$

$$
m_{i|i} = m_{i|i-1} + K_i(x_i - h_i(m_{i|i-1})) \qquad (21)
$$

$$
P_{i|i} = P_{i|i-1} - K_i \bar{H}_i P_{i:i-1} \qquad (22)
$$

The recursive prediction and update steps for EKF are

- Predict:

$$
\bar{Y}_{i-1} = FY_{i-1} \qquad (23)
$$

$$
\bar{P}_{i-1} = FP_{i-1}F^T + Q \qquad (24)
$$

- Update:

$$
K_i = \bar{P}_{i-1}H^T(H\bar{P}_{i-1}H^T + R)^{-1} \qquad (25)
$$

$$
\bar{Y}_i = \bar{Y}_{i-1} + K_i(X_i - H\bar{Y}_{i-1}) \qquad (26)
$$

$$
P_i = \bar{P}_{i-1}(1 - KH) \qquad (27)
$$

In Section IV, we will provide values of different variables used in our experiments.

D. Our Algorithm

We first use the PF algorithm to obtain the user location through the noisy RSSI values. The user location (2D or 3D) obtained using PF is then used as an input into a EKF that reduces the fluctuation in the position estimate resulting in a stable user location estimate. Algorithm 1 summarizes our PFEKF algorithm.

Algorithm 1 Particle Filter-Extended Kalman Filter (PFEKF)

```
procedure PFEKF CASCADE
    Obtain RSSI_recv                         ▷ Obtain RSSI values
    RSSI ← RSSI_recv
    RSSI_filt ← 0                            ▷ Filtered RSSI
    L_i ← (0,0) or (0,0,0)                   ▷ Initialize user location
    while RSSI ≠ 0 do
        L_i ← ParticleFilter(RSSI)
        L_i,filt ← ExtendedKalmanFilter(L_i)
        Print L_i,filt
    end
```

IV. Experimental Setup and Results

To evaluate the performance of our algorithm, we implement an end-to-end prototype that uses the RSSI values of *gimbal* iBeacons to obtain an estimate of the user location. We used our prototype iOS application [13] on an iPhone 6s plus to receive the messages from the iBeacons and retrieve the RSSI values. The user device then forwarded the observed RSSI values to a local Apache Tomcat Server. The particle filtering algorithm running on the server estimated the user's location (the particles with the highest probability are used to obtain the estimate of the user's location). The PF estimated x and y coordinates were then used as input into the EKF algorithm. Figure 1 shows the architecture of our prototype. Table I provides information about the equipment used in our experiment. Below we present the mathematical model for the EKF model used in our experiments.

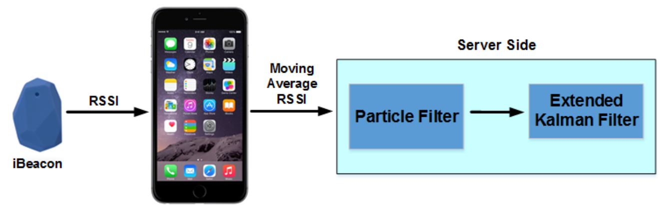

Fig. 1. Our prototype system

TABLE I
SUMMARY OF DEVICE PARAMETERS

Server	Apache Tomcat
Java version	Java 1.8
User Device	Apple iPhone 6s plus
Wireless Interface	Bluetooth V4.2 / 2.4GHz
Operating System	iOS 9.2
Beacons	Gimbal Series 10
Gimbal range	50 meters
Transmission Frequency	100 ms
Major Value	Yes
Minor Value	Yes

A. Mathematical Model for EKF

We use the widely used Position-Velocity (PV) model [19], [20] for EKF modeling. Our state Y_i consists of the current x coordinate, y coordinate, the horizontal velocity V_{xi} component, and the vertical velocity V_{yi} component.

$$Y_i = \begin{bmatrix} x_i & y_i & V_{xi} & V_{yi} \end{bmatrix}^T$$

The state equation for the PV model as given by [20], [19] is given below by Equation (28).

$$\begin{bmatrix} x_i \\ y_i \\ V_{xi} \\ V_{yi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \delta t & 0 \\ 0 & 1 & 0 & \delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{i-1} \\ y_{i-1} \\ V_{xi-1} \\ V_{yi-1} \end{bmatrix} + \begin{bmatrix} m_i^x \\ m_i^y \\ m_i^{V_{xi}} \\ m_i^{V_{yi}} \end{bmatrix} \tag{28}$$

where the matrix given below is the process noise matrix.

$$\begin{bmatrix} m_i^x & m_i^y & m_i^{V_x} & m_i^{V_y} \end{bmatrix}^T$$

Hence the Jacobian matrix F is given by

$$F = \begin{bmatrix} 1 & 0 & \delta t & 0 \\ 0 & 1 & 0 & \delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The parameter δt is the time interval over which the velocity is constant. Since we obtain the measurements after every one second, we used $\delta t = 1$. Similarly the measurement model in Equation (2) can be written as

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ V_{xi} \\ V_{yi} \end{bmatrix} + \begin{bmatrix} n_i^x \\ n_i^y \end{bmatrix} \tag{29}$$

where the Jacobian matrix H is given by

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Parameters P (error covariance), Q (process noise covariance) and R (measurement noise covariance) that are fundamental to accurate performance of the EKF were obtained by trial and error approach in our experiment space and are given below.

$$P = 100I_{44} \quad Q = 0.001I_{44} \quad R = 0.10I_{22}$$

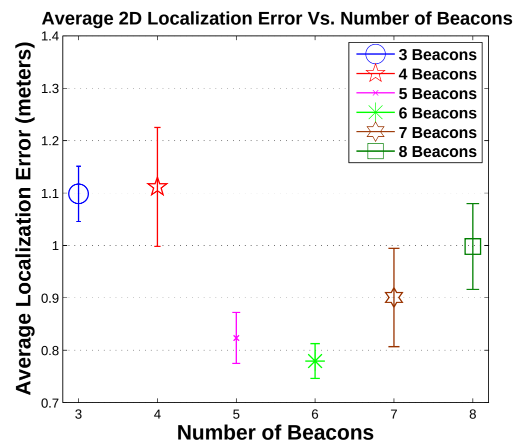

Fig. 2. Average 2D localization error vs number of Beacons in 7m × 6m environment with our PFEKF algorithm on the server side.

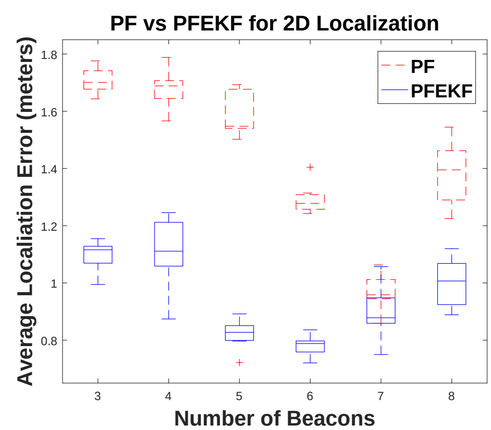

Fig. 3. Average 2D localization error of PF vs. PFEKF in 7m × 6m environment for a different number of beacons

Use of the Extended Kalman Filter involves recursively predicting and updating the state vector as discussed in Section III. We performed our experiments in a 7m x 6m space that contained a number of obstacles such as wooden and metallic cupboards, chairs, tables, and humans, replicating a typical indoor setting. Due to space constraints, the floor plan for the experimental space is relegated to **[add arxiv reference]**. In our experiments, we increased the number of particles in PF from 400 to 2000 with 200 step size. Furthermore, we also varied the total beacons used from three to eight (the point where adding more iBeacons did not improve the performance or deteriorated it in worst case). The beacons were placed on the walls and cupboards, and were 2m above the ground. The phone was placed in a static position at different reference points (points whose location is known) during the experiment. The estimated location obtained through our algorithm was compared with original location of the point. Figure 2 high-

TABLE II
2D LOCALIZATION PERFORMANCE OF PFEKF FOR DIFFERENT NUMBER OF BEACONS IN 7M × 6M ENVIRONMENT.

Particles	**3 Beacons**		**4 Beacons**		**5 Beacons**		**6 Beacons**		**7 Beacons**		**8 Beacons**	
	Mean	**Std**	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**
400	0.995	0.659	0.874	0.385	0.892	0.536	0.836	0.403	1.032	0.480	0.969	0.619
600	1.039	0.579	1.044	0.468	0.838	0.512	0.796	0.384	1.057	0.478	1.086	0.749
800	1.120	0.507	1.246	0.648	0.797	0.485	0.765	0.331	0.750	0.576	1.120	0.447
1000	1.118	0.607	1.222	0.717	0.822	0.471	0.756	0.280	0.920	0.505	1.015	0.524
1200	1.079	0.713	1.209	0.751	0.867	0.614	0.720	0.310	0.889	0.608	1.007	0.598
1400	1.112	0.682	1.111	0.628	0.827	0.559	0.793	0.424	0.849	0.567	1.062	0.532
1600	1.155	0.707	1.128	0.734	0.800	0.423	0.760	0.391	0.878	0.606	0.932	0.634
1800	1.153	0.594	1.109	0.813	0.845	0.479	0.788	0.346	0.862	0.561	0.902	0.482
2000	1.116	0.583	1.064	0.637	0.722	0.359	0.800	0.398	0.869	0.570	0.889	0.517

lights the average 2D localization error for varying number of beacons used in the 7m × 6m environment with server side PFEKF algorithm. In comparison with both PF and KFPF in our prior work [3], [13], the PFEKF has a lower average localization error. Figure 3 compares the PF vs PFEKF for 2D Localization where PFEKF performs better than PF. PF performs optimally in terms of localization error when the number of iBeacons is 7 while PFEKF performs the best with 6 beacons. Hence, PFEKF is more cost efficient as it requires a smaller number of iBeacons.

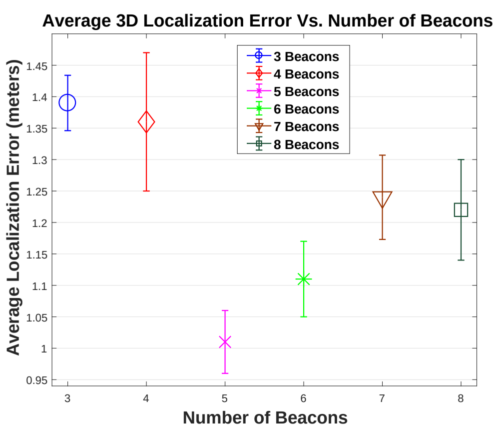

Fig. 4. Average 3D error vs number of Beacons in 7m × 6m environment with our PFEKF algorithm on server side.

Table II provides the average 2D localization error vs. a range of different number of particles using our PFEKF algorithm. The highest localization accuracy was achieved using 6 beacons and 1200 particles. Table III shows the average 3D localization error vs. a range of different number of particles using our PFEKF algorithm. The optimal result was obtained using 7 beacons and 800 particles. It is evident from Figures 3 and 5 that our PFEKF algorithm performs better than using only a PF for RSSI-based indoor localization. In our experiments, the PFEKF improved the localization accuracy by 31.3% and 33.9% in 3D and 2D environments respectively when compared with using only a PF. We also compare PFEKF's performance with KFPF [3]. Figures 6 and 7 compares the performance of KFPF and PFEKF for 2D and 3D localization respectively. While on average the

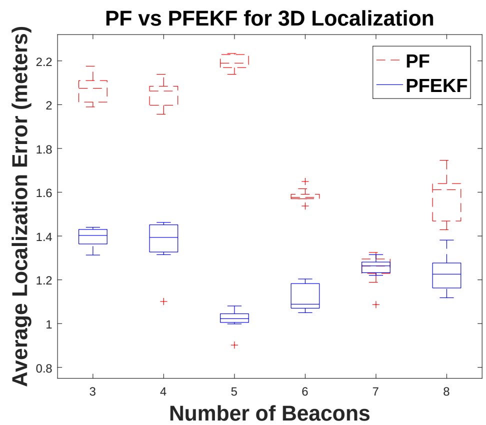

Fig. 5. Average 3D localization error of PF vs. PFEKF in 7m × 6m environment for different number of beacons.

PFEKF outperforms KFPF by 7.6% and 8% in 3D and 2D environments respectively, KFPF performs well if the number of beacons are more than 6. This means that KFPF requires more beacons to perform well because KF is optimal filter for linear models, while indoor localization is non-linear in nature. So increasing the number of beacons provides more reference signals for localization. Furthermore, the lowest localization error is achieved with PFEKF both in 2D and 3D environments, highlighting the fact that PFEKF is preferable over KFPF as KF assumes that the system is linear while in reality, it is non-linear.

V. CONCLUSION

Indoor localization, due to the wide range of applications that it can be provide, has recently seen an increase in interest. While different techniques can be used for indoor localization, RSSI is one of the widely used techniques as it is cost efficient and easy to use. However, the presence of multipath fading and noise in indoor environments affects its performance. In this paper, we proposed the PFEKF algorithm that combines PF and EKF in cascade to enhance accuracy of RSSI-based indoor localization. PFEKF improved the localization accuracy by 31.3% and 33.9% in 3D and 2D environments respectively when compared with using only a PF and by 7.6% and 8% in 3D and 2D environments respectively when compared with using KFPF.

TABLE III
3D LOCALIZATION PERFORMANCE OF PFEKF FOR DIFFERENT NUMBER OF BEACONS IN 7M × 6M ENVIRONMENT.

Particles	**3 Beacons**		**4 Beacons**		**5 Beacons**		**6 Beacons**		**7 Beacons**		**8 Beacons**	
	Mean	**Std**	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**
400	1.336	0.625	1.101	0.356	1.023	0.556	1.089	0.421	1.032	0.480	0.969	0.619
600	1.313	0.590	1.331	0.292	1.081	0.583	1.061	0.404	1.057	0.478	1.086	0.749
800	1.403	0.530	1.462	0.509	0.998	0.465	1.050	0.335	0.750	0.576	1.120	0.447
1000	1.440	0.548	1.451	0.576	1.038	0.522	1.080	0.379	0.920	0.505	1.015	0.524
1200	1.373	0.625	1.453	0.644	1.066	0.638	1.073	0.356	0.889	0.608	1.007	0.598
1400	1.396	0.602	1.351	0.496	1.022	0.582	1.145	0.464	0.849	0.567	1.062	0.532
1600	1.434	0.619	1.394	0.603	1.011	0.407	1.180	0.369	0.878	0.606	0.932	0.634
1800	1.429	0.505	1.403	0.673	1.008	0.526	1.204	0.392	0.862	0.561	0.902	0.482
2000	1.409	0.497	1.315	0.456	0.902	0.410	1.192	0.326	0.869	0.570	0.889	0.517

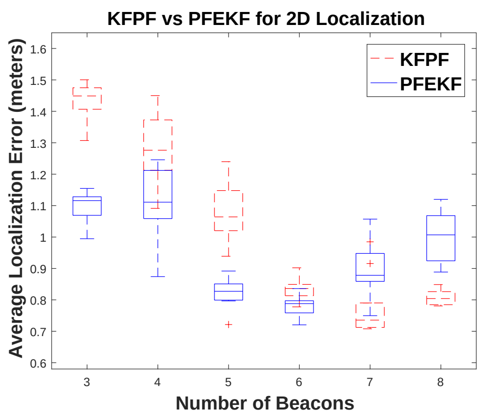

Fig. 6. Average 2D localization error of KFPF vs. PFEKF in 7m × 6m environment for different number of beacons.

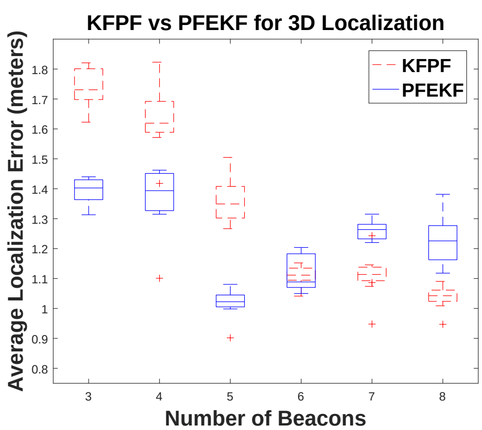

Fig. 7. Average 3D localization error of KFPF vs. PFEKF in 7m × 6m environment for different number of beacons.

REFERENCES

[1] F. Zafari, I. Papapanagiotou, and K. Christidis, "Microlocation for internet-of-things-equipped smart buildings," *IEEE Internet of Things Journal*, vol. 3, pp. 96–112, Feb 2016.

[2] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 37, no. 6, pp. 1067–1080, 2007.

[3] F. Zafari, I. Papapanagiotou, M. Devetsikiotis, and T. Hacker, "An ibeacon based proximity and indoor localization system," *arXiv preprint arXiv:1703.07876*, 2017.

[4] J. Xiong and K. Jamieson, "Arraytrack: a fine-grained indoor location system," in *Presented as part of the 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13)*, pp. 71–84, 2013.

[5] J. Xiong, K. Sundaresan, and K. Jamieson, "Tonetrack: Leveraging frequency-agile radios for time-based indoor wireless localization," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pp. 537–549, ACM, 2015.

[6] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "Spotfi: Decimeter level localization using wifi," in *ACM SIGCOMM Computer Communication Review*, vol. 45, pp. 269–282, ACM, 2015.

[7] P. Kumar, L. Reddy, and S. Varma, "Distance measurement and error estimation scheme for rssi based localization in wireless sensor networks," in *Wireless Communication and Sensor Networks (WCSN), 2009 Fifth IEEE Conference on*, pp. 1–4, IEEE, 2009.

[8] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single wifi access point," in *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*, pp. 165–178, 2016.

[9] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *Proceedings of the 3rd international conference on Mobile systems, applications, and services*, pp. 205–218, ACM, 2005.

[10] I. Guvenc, C. Abdallah, R. Jordan, and O. Dedoglu, "Enhancements to rss based indoor tracking systems using kalman filters," 2003.

[11] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 2, pp. 775–784, Ieee, 2000.

[12] E. Martin, O. Vinyals, G. Friedland, and R. Bajcsy, "Precise indoor localization using smart phones," in *Proceedings of the 18th ACM international conference on Multimedia*, pp. 787–790, ACM, 2010.

[13] F. Zafari and I. Papapanagiotou, "Enhancing ibeacon based micro-location with particle filtering," in *2015 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–7, Dec 2015.

[14] F. Zafari, I. Papapanagiotou, M. Devetsikiotis, and T. J. Hacker, "Enhancing the accuracy of ibeacons for indoor proximity-based services," in *Communications (ICC), 2017 IEEE International Conference on*, pp. 1–7, IEEE, 2017.

[15] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *Signal Processing, IEEE Transactions on*, vol. 50, no. 2, pp. 174–188, 2002.

[16] P. M. Djuric, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Miguez, "Particle filtering," *Signal Processing Magazine, IEEE*, vol. 20, no. 5, pp. 19–38, 2003.

[17] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerospace and Electronic Systems Magazine*, vol. 25, no. 7, pp. 53–82, 2010.

[18] N. Bergman, "Recursive bayesian estimation: Navigation and tracking applications. dissertations no 579," *Linköping Studies in Science and Technology, SE-581*, vol. 83, 1999.

[19] R. Khan, S. U. Khan, S. Khan, and M. U. A. Khan, "Localization performance evaluation of extended kalman filter in wireless sensors network," *Procedia Computer Science*, vol. 32, pp. 117–124, 2014.

[20] R. Khan, F. Sottile, and M. A. Spirito, "Hybrid positioning through extended kalman filter with inertial data fusion," *International Journal of Information and Electronics Engineering*, vol. 3, no. 1, p. 127, 2013.