WiFi Fingerprinting and Tracking using Neural Networks

Peng Wu, *Electrical and Computer Engineering Dept., Northeastern University*, Boston, MA, USA.
Tales Imbiriba, *Electrical and Computer Engineering Dept., Northeastern University*, Boston, MA, USA.
Gerald LaMountain, *Electrical and Computer Engineering Dept., Northeastern University*, Boston, MA, USA.
Jordi Vilà-Valls, *Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO), University of Toulouse*, France.
Pau Closas, *Electrical and Computer Engineering Dept., Northeastern University*, Boston, MA, USA.

BIOGRAPHIES

Peng Wu is a PhD candidate in the Department of Electrical and Computer Engineering at Northeastern University, Boston, MA. His research interests include machine learning with applications to indoor positioning and tracking.

Tales Imbiriba is a Postdoctoral Research Associate in the Department of Electrical and Computer Engineering at Northeastern University, Boston, MA. He received his PhD degree in Electrical Engineering from the Universidade Federal de Santa Catarina (UFSC), Florianópolis, Brazil in 2016. His research interest and background are mainly focused in statistical signal processing, kernel methods, and audio and hyperspectral image processing.

Gerald LaMountain is a PhD candidate and Dean's Fellow in the Department of Electrical and Computer Engineering at Northeastern University, Boston, MA. His current research involves building upon traditional probabilistic techniques for dynamic state estimation and applying those techniques to problems with applications in positioning and localization.

Jordi Vilà-Valls is Associate Professor at ISAE-SUPAERO, University of Toulouse, France. He received the PhD in Signal Processing from INPG, Grenoble, France, in 2010. His primary areas of interest include statistical signal processing, nonlinear Bayesian inference, robust filtering, computational and robust statistics; with applications to localization, tracking and navigation systems.

Pau Closas is Assistant Professor at Northeastern University, Boston, MA. He received the MS and PhD degrees in Electrical Engineering from UPC in 2003 and 2009. He also holds a MS in Advanced Mathematics from UPC, 2014. His primary areas of interest include statistical signal processing, robust stochastic filtering, and game theory, with applications to positioning systems and wireless communications. He is the recipient of the 2014 EURASIP Best PhD Thesis Award, the 9th Duran Farell Award, and the 2016 ION Early Achievements Award.

INTRODUCTION

As technology becomes increasingly interconnected through the advent of advanced telecommunications systems such as 5G networking, the necessity for seamless solutions in location-based services becomes increasingly relevant. In response to this demand, there has been a steady increase in academic and commercial interest in indoor real-time locating systems (RTLSs) and in indoor tracking and positioning. With respect to the broader research domain of Positioning, Navigation and Timing (PNT), indoor positioning and tracking offers a number of difficult challenges. Use of the widely diffused Global Navigation Satellite System (GNSS) technology, one of the most accurate sources of position information when it is available, is often infeasible in indoor or obstructed environments [4]. Instead, alternative systems have to be adopted. One approach to positioning and tracking in such environments is fingerprinting, also referred to as mapping or scene analysis.

The basic idea of fingerprinting is to build a database containing a collection of measured features at designated reference locations within the environment, and to then perform positioning by applying regression techniques to match new measurements to one or more of those in the database. The position in feature space associated through the database with a particular physical reference location is referred to as the "fingerprint" of the environment at that location, and the assumption that these feature vectors are relatively unique forms the basis for the fingerprinting technique. The fingerprinting procedure typically operates in two stages: an offline stage in which the environment is surveyed at known locations and the results are recorded into a database, and an online stage in which navigation is performed by matching new measurements with the content of the database. Once a match is made, position may be inferred based on the reference positions associated with those database measurements.

Fingerprinting offers several important advantages as an indoor positioning technology. One important advantage is that they do not require a specific measurement model: since indoor environments are often segmented and highly non homogeneous in composition, useful observables such as sound and electromagnetic radiation often exhibit highly nonuniform or nonlinear diffuse behaviors which can not be easily or accurately modeled even without advanced and thorough knowledge of the contents of the environment. Fingerprinting replaces the need for an accurate measurement model with a need for an offline training stage, a requirement which may be more realistic in certain contexts. Other advantages are that they are strictly passive and can often make use of existing features of the environment such as installed WiFi networking. For this reason among others, WiFi received signal strength (RSS) fingerprinting is now widely used in indoor positioning and navigation problems [13, 18]. Another remarkable feature of fingerprinting methods is that they do not require any knowledge regarding the location of the transmitting nodes, which is particularly relevant in other schemes such as those which are geometric-based [4].

This contribution presents a novel methodology for performing tracking using RSS fingerprinting. There are two main components in the proposed methodology: *i*) a filtering technique for tracking the receiver given RSS measurements, which is achieved through a Kalman filter algorithm; and *ii*) a neural network based data-driven approach for learn the spatial model for the RSS of the field, which is required in the tracking part if a physics-based model is not used. In this paper we particularly focus on the issues related to RSS field changes, that is when the RSS model is learnt by the neural network and used for tracking the receiver but it changes after some time (e.g. due to people walking in the area, obstacles or walls being added, or other situations causing a change in the RSS. To that aim we compare offline training of the neural network (i.e., where a potentially large training data set is recorded and used for training the model) with online training (i.e., where new data is measured at known locations but is more scarce or not available at the same time so it needs to be processed on a sample-per-sample basis) schemes. In this context, we compare a classical neural network training method (i.e., Adam) and a method based on Kalman filtering, which is suited for sequential inference. To validate the proposed RSS field estimation approaches we simulated an indoor environment using the standard path-loss model and trained offline the model, then generated several disturbances in the model and adapt the neural network to incorporate the new data. Results are provided in terms of training error, tracking error, and tracking convergence rate.

POSITIONING AND TRACKING USING RSS OBSERVATIONS

Fingerprinting is typically conceived as a snapshot approach in which observed features are compared against a pre-built database in order to compute a state estimate. Unlike other state estimation techniques, in fingerprinting methodology the time correlation of the observations due to a time evolution of the state is seldom used. In certain solutions, there is also a filtering or smoothing of those independent snapshot estimates. For applications where the state is time-varying, the task is often best formulated as a tracking problem which, given an accurate model, can be solved through a variety of approaches which leverage both linear and nonlinear algorithms. One important application of both fingerprinting and tracking is in positioning, where the unknown location of a user or object is estimated from direct or indirect observations. When radio frequency (RF) measurements are available, RSS may be used for this observable. Our goal in this contribution is to develop a spatial positioning methodology based on RSS observables which leverages both fingerprinting and tracking techniques, and consequently overcomes the individual limitations of each approach. Namely, that fingerprinting is generally used in static estimation and that tracking algorithms rely on a well defined analytic model.

Following the traditional approach for solving tracking problems, we formulate the underlying system in terms of a state space model. Depending on the specific positioning task being performed, this may include velocity or other parameters of interest in addition to the spatial location of the object being tracked. For the purposes of this paper we will consider an object moving through an RSS field according to a 2-dimensional 2$^{\text{nd}}$ order kinematic model. The state is defined as

$$\mathbf{x}_t = [p_x^{[t]}, p_y^{[t]}, \dot{p}_x^{[t]}, \dot{p}_y^{[t]}]^\top ,$$

with $p_\zeta^{[t]}$ being the position at time t in the ζ-axis and $\dot{p}_\zeta^{[t]}$ its derivative with respect to time. In tracking, one is interested in learning the posterior distribution of that state given all available information, $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ where $\mathbf{y}_t$ are the measured RSS in decibels at time t for a set of L reference nodes $\mathbf{y}_t = [y_1^{[t]}, \dots, y_L^{[t]}]^\top$. Applying these definitions for the state and observations, the state space model is formulated as

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}) + \boldsymbol{\epsilon}_t^x \tag{1}$$
$$\mathbf{y}_t = \mathbf{h}(\mathbf{x}_t; \mathbf{w}_t) + \boldsymbol{\eta}_t \tag{2}$$

where the *known or estimated* state evolution and measurement functions are denoted $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$ respectively. In the RSS positioning application $\mathbf{h}(\cdot)$ is an injective function modelling the relationship between the average RSS measured and a given location. Since in general this function will not be known for a given RSS field, our approach will be to consider the use of a neural network to estimate that relationship. This results in a model which is parameterized by the weights of a neural network, $\mathbf{w}_t$, so the function $\mathbf{h}$ is the forward propagation of the neural network. This function will be substantially nonlinear, and as a consequence, only suboptimal tracking methods can be considered when inferring $p(\mathbf{x}_t|\mathbf{y}_{1:t})$. Regarding the state evolution function, for the sake of simplicity, it is considered as a linear mapping (i.e., $\mathbf{f}(\mathbf{x}) = \mathbf{F}\mathbf{x}$) in this paper [4]

$$\mathbf{F} = \begin{pmatrix} \mathbf{I} & \Delta T \cdot \mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} , \tag{3}$$

where ΔT is the time between consecutive time instants. In this contribution we consider a standard Extended Kalman filtering (EKF) to estimate the posterior distribution of interest, which basically involves linearizing the measurement function $\mathbf{h}$. Finally, we approximate the additive process and measurement noise $\boldsymbol{\epsilon}_t^x$ and $\boldsymbol{\eta}_t$ as being Gaussian random variables with zero mean and known or estimated covariance.

The recursion for computing the predicted distribution $p(\mathbf{x}_t|\mathbf{y}_{1:t-1})$ and the filtering distribution $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ at the time step t is given by the Bayesian filtering equations [15]. In particular, given an initialization or prior distribution of the states $p(\mathbf{x}_0)$, the posterior distribution can be conceptually obtained in two steps: prediction and update. In the first of these two steps, the predictive distribution is computed using the Chapman–Kolmogorov equation for the step t,

$$p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1}) p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1}. \tag{4}$$

and updates it with the latest measurement $\mathbf{y}_t$ to compute the posterior distribution using Bayes' rule

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) = \frac{1}{z_t} p(\mathbf{y}_t|\mathbf{x}_t) p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) \tag{5}$$

where the normalization constant z_t is given as

$$z_t = \int p(\mathbf{y}_t|\mathbf{x}_t) p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) d\mathbf{x}_t \ . \tag{6}$$

The main challenge in performing this recursion is solving the two integrals in (4) and (6), for which closed-form solutions only exist in limited cases (e.g. when the state-space model is linear and Gaussian). In instances where the nonlinearity is constrained to a given order, it is possible to leverage established techniques for performing tracking a nonlinear model. Whereas the EKF relies on linearizing the model by analytically or numerically computing matrix derivatives of $\mathbf{f}$ and $\mathbf{h}$ with respect to the state variable $\mathbf{x}_t$, the Cubature Kalman filtering (CKF) leverages spherical-radial cubature rules to compute an approximation of the propagation of a Gaussian random variable (i.e. the state) through these nonlinear functions [1].

DATA-DRIVEN RSS FIELD LEARNING

Along with other techniques, the use of neural networks has been widely considered for indoor positioning problems in fingerprinting scenarios. The literature presents a plethora of neural network-based algorithms which include convolutional neural networks (CNN) for classification of RSS measurements [7, 16], deep neural networks [5, 9, 16], RBF networks [11], standard multi-layer perceptrons (MLPs) [2], and general regression neural networks (GRNN) [8]. The vast majority of neural network-fingerprinting-based methods aim at learning functions that directly map measured RSS values to their respective target locations. Although accurate position estimates are provided by considering such approaches in a static scenario, the inability of traditional fingerprinting methods to adapt to a dynamical environment is a major drawback when considering real world applications. Therefore, the demand for adaptive strategies, capable of matching the dynamical behavior of different scenarios, arises naturally in this context and remains still an open problem [4].

The literature contains several works which combine an offline stage using fingerprinting with some kind of adapted tracking strategy. Some of these present strategies which include Viterbi-like algorithms [4], although concerning the use of neural networks in this context a small number of works can be found combining those with tracking schemes. Recently, the authors in [8] applied a GRneural network to learn the fingerprinting map (RSS to position) and used a Kalman filter to improve the GRneural network position estimates providing smoother results without discontinuities obtained when the GRneural network are directly used to predict positions over time (tracking). It is worth noticing that our goal is to present a methodology where both fingerprinting and tracking components of the RTLS are intertwined such that the overall performance is enhanced.

Our approach aims at providing such interrelated system where fingerprinting and tracking methodologies are tight together through the use of neural networks and variants of the Kalman filter. For this, we invert the usual fingerprinting formulation by learning the RSS fields from position inputs. This allows one to formulate the training and tracking problems as a state space model which can be solved sequentially. To achieve this goal, we start by considering two different possible strategies using fingerprinting. In the first strategy, the network is trained using standard gradient-based procedures in the offline and online stage while tracking is performed using KFs. This strategy results in two disjoint steps to solve the tracking problem. In a second scenario, we propose to train the neural networks using a KF formulation [6]. This, potentially, unifies the learning and tracking methodology which are both, now, formulated in the context of (nonlinear) Kalman filters. This formulation allows us to easily scale the resulting algorithm to the availability of new fingerprinting data, thus, allowing the system to adapt to changes in the environment and the availability of new training data.

For the RSS field learning task we consider traditional multilayer perceptron (MLP) neural networks [2]. In our solution the neural network structure is envisioned to accept position vectors as inputs and present estimated RSS vectors as outputs.

Neural network training

In both settings (single and multiple neural networks) we consider both gradient-descent-based and Kalman filter strategies to adjust the neural networks coefficients. In the gradient-based methods we consider two well known strategies: stochastic gradient-descent (SGD) and Adaptive Moment Estimation (Adam) [10]. SGD is by far the most popular optimization strategy used in neural network training. The algorithm is very efficient, although it has the potential to become stuck in local minima [12]. The weight adaptation is provided by an update in the direction that minimizes the prediction error, driven by the gradient of the error,

$$\mathbf{w}_t = \mathbf{w}_{t-1} - \gamma \nabla_{\mathbf{w}} Q \tag{7}$$

where it becomes clear that $\mathbf{w}_t$ is update from $\mathbf{w}_{t-1}$ in the direction that minimizes the gradient. The γ is the learning rate, determined the step size of gradient descent. $\nabla_{\mathbf{w}} Q$ is the derivative of quadratic error function over the current weights. On the other hand, Adam is a method for efficient stochastic optimization that, differently to many of its counterparts, only requires first-order gradients. The optimization procedure computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradient, thus, promoting some kind of step size annealing [10]. Excellent results have been shown in the literature when modeling stochastic systems what makes Adam a suitable strategy for learning RSS fields in the context of the present problem.

Alternatively, KF training of neural networks was introduced as a second-order neural network training method that is a practical and effective alternative to the batch-oriented, second-order methods such as quasi-Newton, Levenburg-Marquardt, and conjugate gradient strategies [6]. EKF [15], and its more flexible variants such as Unscented KF (UKF) and CKF [15] have been considered to neural network training over the years in different fields [6, 17]. In this work, we consider EKF as the training strategy for neural networks, thus, proposing a unified Kalman-based strategy for the online and offline stages of the fingerprinting methodology. When using Kalman filter to train a neural network, the training strategy is interpreted as a state estimation problem where the weights of the neural network are treated as a random process which is partially observed from labeled input/output training data, $\mathcal{D}_T = \{\mathbf{p}_t, \mathbf{y}_t\}_{t=1}^{T}$. Each time a new training pair is processed, the weight vector will be updated. The state space model is

$$\mathbf{w}_t = \mathbf{w}_{t-1} + \boldsymbol{\epsilon}_t^w \tag{8}$$

$$\mathbf{y}_t = \mathbf{h}(\mathbf{w}_t; \mathbf{p}_t) + \boldsymbol{\eta}_t \tag{9}$$

where the measurement function $\mathbf{h}(\cdot)$ is the same as in (2). The main difference in (9) is that $\mathbf{p}_t$ are known inputs and $\mathbf{w}_t$ is the independent variable, whereas in (2) it is the other way around and the weights are considered known inputs defining the model. Regarding the evolution of the weights, we assume a static dynamic affected by a random defined as $\boldsymbol{\epsilon}_t^w$ and statistically characterized by a normal distribution whose variance models the amount of variability we allow in the weights on every time step.

Regarding some implementation details, we highlight the computation of the Jacobian matrix over the MLP weights and the setting of the EKF parameters. The Jacobian matrix, which is needed by both gradient-based and EKF strategies, is computed using *Automatic Differentiation* by creating Computational Graphs [3], while the EKF state and modeling errors covariance matrices were initialized as diagonal matrices such as $\zeta\mathbf{I}$ where ζ is a small positive constant.

Offline vs Online learning of RSS fields

One scenario that we are particularly interested in addressing is that of time-varying RSS fields. In a real-world scenario, it is very likely that the environment will change. For example, furniture might be added in the surveyed area, the number of persons interacting with the environment at any given time might change, or other situations might occur which can change the learned RSS spatial field either partially or in its totality. In other words, any change in the environment will likely have an impact on the received signal and its characteristics. If we insist in using the (neural network-based) model that was trained before the changes in the environment, the results might be erroneous or biased. Therefore, it seems reasonable to allow the model to update as new data is available.

In this work, we split the training process into two phases: an offline phase, where data initially collected under an static environment is used to learn the RSS fields; and online phase, where we assume the environment to be time-varying over space. Thus, the model initially learned in the offline setting must be adapted to cope with changes in the environment. Since the field may have changed along with changes to the environment, the initial data set no longer provides an appropriate characterization of the field and, in the online phase, only new acquired samples must be used. Furthermore, the possibility that the field may be continuously changing makes the batch training used in conventional MLP training not suitable.

The transition between the offline and online phases is transparent for the KF-based strategy due to the sequential nature of Bayesian filtering methods. For the Adam, however, the training in the offline and online phases may differ due the possibility of using data batches and many epochs in the first scenario and not in the later. Thus, in this work, we train the MLP using Adam in different ways during the online and offline phases. During the offline phase we consider $N_e^{\text{offline}} >> 1$ epochs with batch size of $N_b^{\text{offline}} >> 1$ samples. During the online phase, however, we perform just one iteration of Adam for each new sample, that is $N_e^{\text{online}} = N_b^{\text{online}} = 1$. By doing so, we avoid the risk of mixing samples that characterize different environment states.

EXPERIMENTAL VALIDATION

In this section we use synthetic data to test the discussed strategies. Following a standard fingerprinting-based procedure [4], we assume that an initial data set $\mathcal{D}_0 = \{\mathbf{p}_i, \mathbf{y}_i\}_{i=1}^N$ of locations $\mathbf{p}_i \in \mathbb{R}^2$ and a vector of RSS values $\mathbf{y}_i \in \mathbb{R}^L$, with L being the number of access points, are available. To simulate the RSS field, initially, we use the standard path loss model [14], which we assume static during the data collection. To observe the behavior of the discussed learning methodologies under an online learning environment we simulate a field change after the initial training phase and observe the performance of the algorithms as new measurements are collected.

As a performance metric, used for both field estimation and indoor tracking, we compute root mean-square error (RMSE). For the RSS field estimation the RMSE is given by

$$\text{RMSE}_{\mathbf{y}} = \sqrt{\frac{1}{NL}\sum_{i=1}^{N}\|\mathbf{y}_i - \hat{\mathbf{y}}_i\|^2} \tag{10}$$

with $\mathbf{y}$ and $\hat{\mathbf{y}}_i$ being the true and estimated RSS value at location $\mathbf{p}_i$, respectively. For the tracking performance analysis we use the $\text{RMSE}_\mathbf{p}$ between the generated trajectory position $\mathbf{p}_t$ and the estimated positions $\hat{\mathbf{p}}_t$. The appropriate $\text{RMSE}_\mathbf{p}$ expression can be obtained by replacing $\mathbf{y}$'s by $\mathbf{p}$'s in (10), and N and L by appropriate values. Since the tracking algorithm may diverge, specially if the field estimation is poor, the $\text{RMSE}_\mathbf{p}$ is computed only for the non-divergent trajectories.

Monte Carlo runs were performed to access the mean-behavior of the field learning and tracking procedures. In such cases, the proportion of non-divergent tracking evaluations is also presented.

Data Generation

To generate the simulated scenario we considered a 50×20 m^2 room with $L = 7$ identical access points placed in known arbitrary locations $\mathbf{p}_a^{[\ell]}$, $\ell = 1, \dots, L$. The RSS field generated by each access point at an arbitrary location $\mathbf{p}_i$ is given by

$$y_i^{[\ell]} = P_T - P_0 - 10\lambda \log_{10} \frac{d(\mathbf{p}_i, \mathbf{p}_a^{[\ell]})}{d_0} + \eta_i^{[\ell]} \tag{11}$$

where, d_0 is a reference distance, P_0 is the attenuation at such distance in dB, $d(\mathbf{p}, \mathbf{p}')$ is the Euclidean distance between positions $\mathbf{p}$ and $\mathbf{p}'$, P_T is the transmitted signal power in dBm, and $\eta_i \sim \mathcal{N}(0, \sigma_\eta^2)$ is an additive WGN. The i-th vector of measured RSS is then obtained by stacking the L values as $\mathbf{y}_i = [y_i^{[1]}, \dots, y_i^{[L]}]^\top$. λ denotes the path loss exponent. For all simulations we considered $P_T = 1$, $P_0 = 20$, $\lambda = 3$, $d_0 = 0.1$, and $\sigma_\eta^{[\ell]} = 4$. To construct the initial data set, we sampled the space using a grid with resolution of 1 m, resulting in a total of $N = 1071$ samples. Figure 1, left panel, shows the initial simulated RSS fields where the peaks of maximum RSS happen at the AP locations $\mathbf{p}_a^{[\ell]}$.

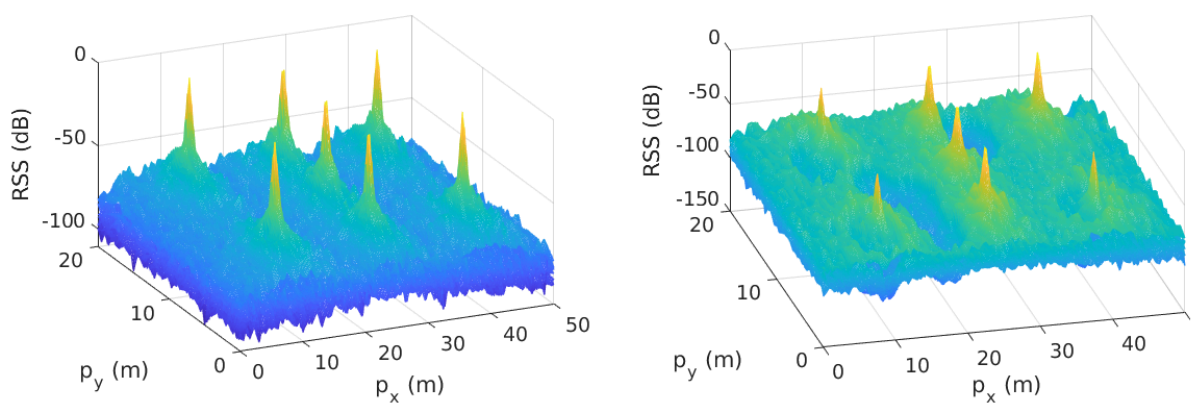

Figure 1: RSS fields for all access points in a 50×20 m^2 room. Initial noisy fields (left), and the field when using the bump functions (right).

In order to evaluate the ability of the neural network to adapt to changes in the fields, we perturbed the field in Fig.1 in the online training process. To simulate field changes we follow three different approaches which vary in the amount of resulting field difference and complexity of the new field. The simplest and most mild field change was performed by changing the λ in the path loss model. Specifically, we considered $\lambda_{\text{new}} = 5$, implying a faster attenuation over distance. The second approach consisted in changing the location of two access points, moving the two access points on the edge further away from where they were. The third approach consisted in modeling randomly positioned smooth field changes over space. To this end, we considered bump-shaped attenuation functions to simulate RSS attenuation caused by large objects or walls in the room. The bump-shaped function is given by

$$\varphi_\mathbf{c}(\mathbf{p}) = \begin{cases} \xi \exp\left(-\frac{1}{r - \|\mathbf{p}-\mathbf{c}\|^2}\right) & \text{, if } \|\mathbf{p}-\mathbf{c}\|^2 < r \\ 0 & \text{, otherwise} \end{cases} \tag{12}$$

where $\xi > 0$ controls the maximum function amplitude, r is the radius of the bump, and $\mathbf{c}$ is the location were the function is centered. For simplicity, we assumed that the attenuation caused by the bump function is the same for all access points. Then, assuming N_b randomly centered bump-functions, the RSS model becomes

$$y_i^{[\ell]} = P_T - P_0 - 10\lambda \log_{10} \frac{d(\mathbf{p}_i, \mathbf{p}_a^{[\ell]})}{d_0} - \sum_{j=1}^{N_b} \varphi_{\mathbf{c}_j}(\mathbf{p}_i) + \eta_i^{[\ell]}. \tag{13}$$

Clearly, this third approach adds more complexity to the resulting RSS field. In the simulations with the bump function attenuation we considered $\xi = 10$, $N_b = 20$, $r = 10$. The RSS fields generated using (13) are presented in the right panel of Figure 1.

To simulate an online scenario we assume that new RSS vectors $\mathbf{y}_t^{\text{new}}$ are randomly sampled from the space. As new samples arrives we update the MLPs using both EKF and Adam. To access the MLP training convergence in both offline and online scenarios two test data set were created, one for each scenario, using 1000 samples randomly from space.

Discussion

All simulations presented in this section are composed of an offline phase and an online phase. The offline phase uses noisy data generated using the path loss model as discussed in the previous section. The beginning of the online phase is marked by a field change after which algorithms which were trained offline need to re-adapt. While the offline and online phases are essentially the same for KF strategies, for conventional gradient methods this situation presents a challenge since samples are collected sequentially rather than in batch. During the offline phase we trained and tested MLPs using Adam for data sets composed of 20, 100 200, 500, and 1071 (all) samples using 20 epochs. Due to the recursive nature of Kalman filters we tested the EKF performance for all possible data subset sizes ($N \in [1, 1071]$). In the online phase Adam is trained sequentially using one sample at a time.

Figures 2, 3 and 4 present the evolution of the $\text{RMSE}_{\mathbf{y}}$ in (dB) for both MLP training methods, Adam and EKF, as the amount of available data grows and for offline and online scenarios. The field changes which occur at sample 1071 are clearly visible in all figures as an increase of the RMSE, and this transition delimits the beginning of the online training phase. In all three scenarios, the results of the offline training phase are very similar. In each case, the MLPs trained with the Kalman filter have faster initial convergence but as the amount of data available for training increases, eventually Adam slightly outperforms the EKF strategy. Conversely, the performance of the online phase differs in all three scenarios as the complexity of the field changes vary. In each case however, the EKF clearly outperforms Adam in the online scenarios, presenting much faster convergence and better overall performance. This result is expected due the natural recursive and optimal design of Bayesian filters, and the fact the Adam when used for online training in this way is unable to iteratively improve over many epochs with randomized data batches.

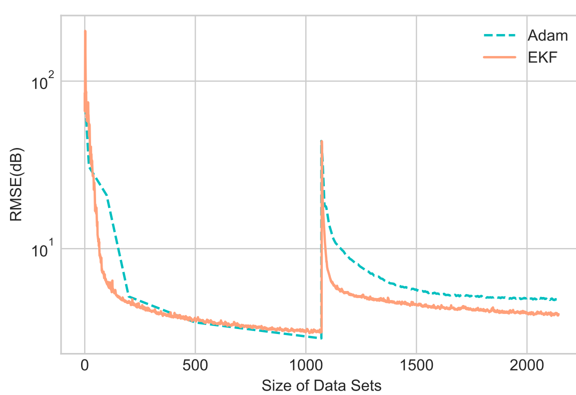

Figure 2: RMSE training error in offline and online phase (path loss exponent change from $\lambda = 3$ to $\lambda = 5$), trained with Adam and EKF methods.

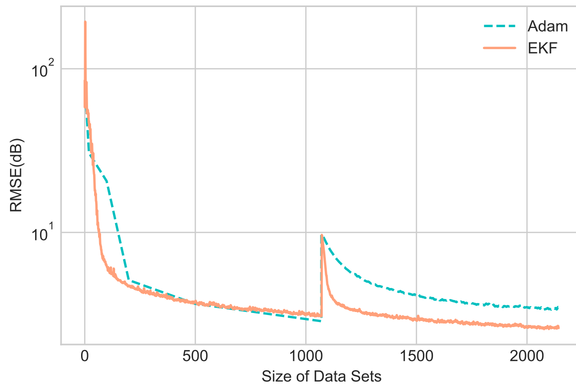

Figure 3: RMSE training error in offline and online phase(Two access points are moved), trained with Adam and EKF methods.

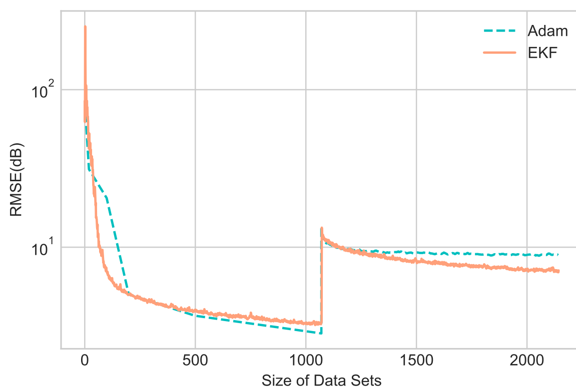

Figure 4: RMSE training error in offline and online phase(bumps are added), trained with Adam and EKF methods.

To assess the performance of the training algorithm in an indoor tracking context, we used the KF-based tracking strategy discussed in Section 2. We assume $\mathbf{h}(\cdot)$ to be estimated using the MLPs trained with Adam and the EKF. To provide a performance baseline we also considered the case where $\mathbf{h}(\cdot)$ was taken as the true underlying path loss model used to build the measurements. The results of this experiment presented in Figures 5 to 7 are consistent with the results of the online phases of Figures 2 to 4 respectively. The tracking performance was evaluated as the MLPs were learning the field changes. Specifically, we tested the tracking performance after 0 (the moment the field changed), 20, 100, 200, and 1000 data samples of the online learning algorithms. For each combination of data set size and field change, $N_r = 100$ Monte Carlo runs were made with trajectories generated performing a Gaussian random walk using the 2-D kinematic model detailed in (1) with $\Delta T = 0.25$ seconds. The figures present the $\text{RMSE}_{\mathbf{p}}$ results as well as the percentage of non-divergent tracking evaluations.

For Figures 5 and 6, the changes to the field are accounted for in the path loss model as a change in parameters. Consequently, the path loss model presents the best tracking performance, since it accurately represents the true field. This is not the case in Figure 7 since the attenuation using the bump-functions can not be characterized by the path loss model.

Concerning the MLP-based methods, it is clear that as the algorithms converge, the tracking performance improves with respect to both $\text{RMSE}_{\mathbf{p}}$ and the percentage of non-divergent tracking solutions. This is specially clear in Figures 5 and 3 where the field changes were mild. In both cases superior performance was observed when using the EKF-MLP, with an average precision of less than 2 meters after 1000 samples were observed, however, with only 100

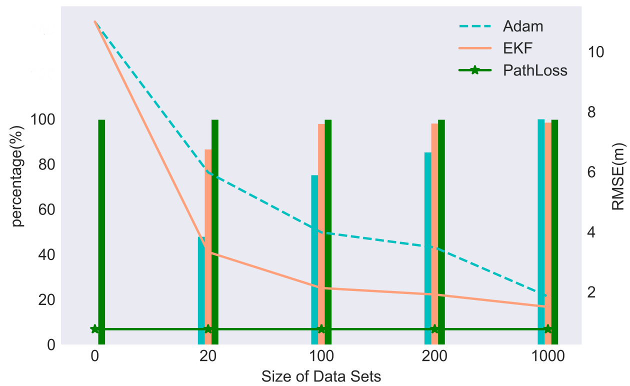

Figure 5: Online phase tracking error and percentage after retraining the neural network with0,20,100,200,1000 data samples (λ changes from 3 to 5)

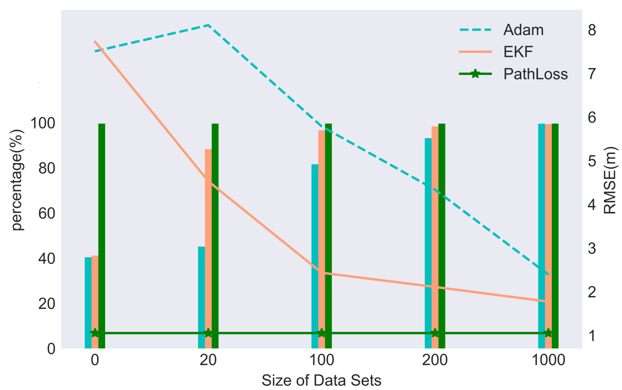

Figure 6: Online phase tracking error and percentage after retraining the neural network with 0,20,100,200,1000 data samples (2 access points are moved)

samples similar performance was observed with both methods.

Figure 7, presents the biggest challenge for all methods since it corresponds the most dramatic change to the field. In this scenario, all methods provided worse performance, with the Adam and EKF behaving similarly in terms of $\mathrm{RMSE}_{\mathbf{p}}$ up to 100 data samples. For 200 and 1000 samples however, the EKF provides the best results. This results is consistent with the results of the online phase of Figure 4, which shows both methods presenting a slow convergence rate in this more complex scenario, ultimately showing a similar initial convergence that grows as the number of observed samples increases.

CONCLUSION

This paper presents a methodology to use tracking solutions seamlessly in the context of RSS fingerprinting. We reversed the traditional fingerprinting formulation which directly estimate position vectors from RSS measurements. In contrast, we use a data-driven machine learning method to learn the RSS field at any location. The reverse formulation allows us to implement a solution, based on the KF machinery, to track the location of a receiver based on RSS measurements and the reconstructed RSS field model. The overall methodology results in a unified KF-based strategy for both stages of fingerprinting indoor tracking problem, which is seen to be highly beneficial in terms of online learning as the conditions of the scenario change.

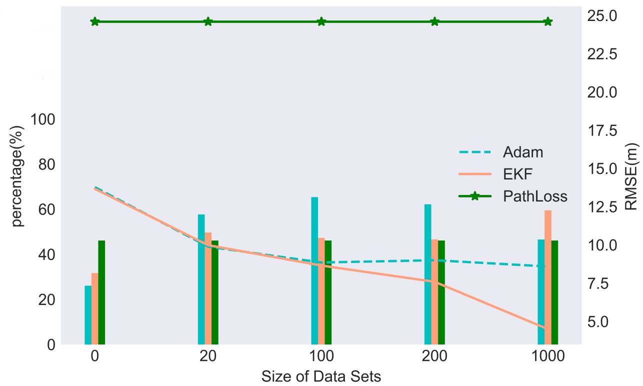

Figure 7: Online phase tracking error and percentage after retraining the neural network with 0,20,100,200,1000 data samples (bumps are added)

Simulations with synthetic data were used to demonstrate the performance of the KF-based MLP learning strategy under offline and online scenarios in contrast to gradient-based algorithms such as Adam. Although the EKF training method resulted in faster convergence and improved tracking capability in all scenarios, the simulations shows that Adam can still outperform the EKF in a batch setting if the data set is large enough. This, however, is rarely the case, specially in dynamical environments where small amount of data may be collected under a stationary time window in which case sequential training schemes might be more desirable to consider.

ACKNOWLEDGEMENTS

This work was partially supported by the National Science Foundation under Awards CNS-1815349 and ECCS-1845833.

REFERENCES

[1] I. Arasaratnam and S. Haykin. Cubature Kalman filters. *IEEE Trans. Automatic Control*, 54(6):1254–1269, June 2009.

[2] R. Battiti, N. T. Le, and A. Villani. Location-aware computing: a neural network model for determining location in wireless lans. Technical report, University of Trento, 2002.

[3] Y. Bengio et al. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[4] D. Dardari, P. Closas, and P. M. Djuric. Indoor tracking: Theory, methods, and technologies. *IEEE Transactions on Vehicular Technology*, 64(4):1263–1278, 2015.

[5] G. Félix, M. Siller, and E. N. Alvarez. A fingerprinting indoor localization algorithm based deep learning. In *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 1006–1011. IEEE, 2016.

[6] S. Haykin. *Kalman filtering and neural networks*, volume 47. John Wiley & Sons, 2004.

[7] M. Ibrahim, M. Torki, and M. ElNainay. CNN based indoor localization using RSS time-series. In *2018 IEEE Symposium on Computers and Communications (ISCC)*, pages 01044–01049. IEEE, 2018.

[8] S. R. Jondhale and R. S. Deshpande. Kalman filtering framework-based real time target tracking in wireless sensor networks using generalized regression neural networks. *IEEE Sensors Journal*, 19(1):224–233, 2019.

[9] K. S. Kim, S. Lee, and K. Huang. A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting. *Big Data Analytics*, 3(1):4, 2018.

[10] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[11] C. Laoudias, P. Kemppi, and C. G. Panayiotou. Localization using radial basis function networks and signal strength fingerprints in wlan. In *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference*, pages 1–6. IEEE, 2009.

[12] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[13] G. M. Mendoza-Silva, P. Richter, J. Torres-Sospedra, E. S. Lohan, and J. Huerta. Long-Term WiFi Fingerprinting Dataset for Research on Robust Indoor Positioning. *Data*, 3(1), 2018.

[14] A. Moragrega, P. Closas, and C. Ibars. Potential game for energy-efficient RSS-based positioning in wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, 33(7):1394–1406, 2015.

[15] S. Särkkä. *Bayesian filtering and smoothing*, volume 3. Cambridge University Press, 2013.

[16] X. Wang, L. Gao, S. Mao, and S. Pandey. CSI-based fingerprinting for indoor localization: A deep learning approach. *IEEE Transactions on Vehicular Technology*, 66(1):763–776, 2017.

[17] Y. Wang, S. Chai, F. Khan, and H. D. Nguyen. Unscented Kalman Filter trained neural networks based rudder roll stabilization system for ship in waves. *Applied Ocean Research*, 68:26 – 38, 2017.

[18] S. Xia, Y. Liu, G. Yuan, M. Zhu, and Z. Wang. Indoor fingerprint positioning based on Wi-Fi: An overview. *Isprs International Journal Of Geo-Information*, 6(5), 2017.