Dictionary learning based fingerprinting for indoor localization

Chirag Kumar, Ketan Rajawat
Dept. of Electrical Engineering
Indian Institute of Technology Kanpur
Kanpur-208016,India
Email: {chiragar,ketan}@iitk.ac.in

***Abstract*—Indoor localization is often challenging due to the non-availability of GPS signals. Recently, various radio frequency fingerprinting techniques have been proposed to identify indoor locations using simply received signal strength (RSS) measurements. In general however, RSS measurements are time-varying and are difficult to model for complex environments. This paper proposes the use of dictionary learning (DL) to generate high quality fingerprints that depend also on the channel characteristics for each location. An enhanced DL algorithm is proposed that utilizes prior information about the channel distribution, and can generate the fingerprints in an online fashion. Simulation results demonstrate the efficacy of the proposed approach.**

I. Introduction

Location information plays an important role in a number of applications, such as environment monitoring, surveillance, mining and healthcare based applications. A number of mobile applications also rely on location information to provide various services to the users. However, obtaining absolute or even relative coordinates becomes challenging in indoor or shadowed environments, where GPS signals cannot reach [1]. The recent trend is to utilize the already existing wifi and cellular infrastructure for indoor localization and positioning [1]–[5].

The problem is quite challenging and attracted lot of researchers resulting in various methodologies. Commonly used method are triangulation [6], [7], proximity based methods. However these methods are complex and require line of sight communication to find the location.

A commonly employed approach is the so-called fingerprinting, where each location is identified by some kind of features. Fingerprinting is done in two steps: i) training phase, where features corresponding to unknown area are collected and stored in a database ii) mapping phase, current feature is matched with the existing fingerprints and the one having maximum similarity is the required location. Generally used fingerprints are generated using the received signal strength from various access points [8], [9]. The use of RSS for fingerprinting provides various advantages such as simple to model and works well with existing framework. The approach works well for static environment i.e. same access points(AP) are available all over the area, where a large number of APs are installed at fixed locations, transmitting at fixed power. The approach is however difficult to extend to environments where access points are sparse or operate intermittently. More generally, location fingerprinting must utilize the available cellular signals as well, which have relatively denser coverage [10]. However, cellular signals are dynamic, subject to path loss and fading impairments, and generally cannot be demodulated and decoded. This motivates the use of training-based fingerprinting techniques that can handle time-varying transmitter activity using RSS measurements alone.

One of the popular approaches entail propagation modeling and subsequent estimation of the relevant parameters. For instance, the work in [10]–[12] models the RSS using a simple path loss model, and learn all its parameters during the training phase. Such a model-based approach however does not work under channel impairments, such as shadowing and fading. While channel variations can simply be averaged out over a long duration, it amounts to discarding the location-specific channel characteristics. Ideally, channel information must be utilized to obtain better quality fingerprints; see e.g. [10] and references therein.

This paper advocates a model-free approach to location fingerprinting via DL [13], [15]. More precisely, the received signal strengths at each location are expressed as sparse linear combinations of several atoms or basis vectors, and the dictionary of these atoms constitutes the unique fingerprint for that location. The DL algorithm discards all the ”temporary” features of RSS measurements, and can therefore be used in time-varying scenarios. In the context of wireless communications, the dictionary can also be interpreted as the matrix of channel characteristics, and the sparse coefficients as transmit powers [16], [17]. Existing DL-based localization techniques such as the one in [18] differs from the present work as it utilized the sparse vectors for localization. In contrast, the DL framework utilized here adopts a classification-based approach [19], [20] that is fundamentally different. Localization via classification has been considered before, where support vector machines (SVM) [21], k-nearest neighbors(kNN) [22], and neural networks classifiers [23] were utilized. It is remarked that the present approach using DL is comparably more flexible, and can not only handle missing information but can also be implemented in an incremental or online fashion.

As a second contribution, this paper extends the interpretation by modifying the DL algorithm to incorporate prior information regarding the channel distribution. An online algorithm is proposed that allows the dictionary to be learned

978-1-5386-1224-8/18/$31.00 ©2018 IEEE

as measurements arrive. Simulations are performed to demonstrate the efficacy of the proposed framework.

This paper is organized as follows: Sec. II provides an overview of system model and background of DL. In Sec. III algorithm for DL based fingerprinting is presented. Simulation Results have been discussed in Sec. IV followed by conclusion in Sec. V.

Notation: Before proceeding to the system model and background, some common notations are introduced. Kronecker product is denoted by $\otimes$, while transpose is denoted by ^{T}. Bold upper (lower) case letters denote matrices (vectors), whose sizes are not stated if they are clear from the context. The (i, j)-th entry of a matrix $\mathbf{X}$ is denoted by $[\mathbf{X}]_{ij}$.

II. System Model and Background

Consider a network with an unknown number of transmitters, transmitting at a total of F frequencies, indexed by $\{1, 2, \ldots, F\}$. At time t, let y_{ft} be the RSS observed by a particular receiver at frequency f. Given $\{y_{ft}\}_{f,t}$, the goal is to obtain a RSS-specific fingerprint for the receiver's current location.

Classically, location fingerprints are generated by observing the RSS at each location, and utilizing the path-loss model [?], [10]–[12]. The RSS at a given frequency and time is modeled as

$$y_{ft} = p_{ft} L_p(d_0) \left(\frac{d}{d_0}\right)^{-\alpha} + e_{ft} \tag{1}$$

where p_{ft} is the transmit power, α is the path-loss coefficient that depends on the environment, $L_p(d_0)$ is pathloss at the reference distance d_0, and d is the distance between the transmitter and the receiver. The error e_{ft} is generally assumed to be normally distributed, with unknown variance. Within the framework considered here, the parameters $\{p_{ft}\}$, $L_p(d_0)$, d, and α are also unknown, and must be learned from the RSS measurements. Since the some of these parameters may be location or time-dependent, the path-loss model may not provide good location-specific fingerprints.

This paper considers a blind, model-free, RF fingerprinting technique that utilizes dictionary learning. The DL approach postulates the following equation for the RSS at frequencies $\{1, \ldots, F\}$:

$$\mathbf{y}_t = \mathbf{G}\mathbf{p}_t + \mathbf{e}_t \qquad t = 1, 2, \ldots, T \tag{2}$$

where $\mathbf{y}_t$ collects $\{y_{ft}\}_{f=1}^F$. Here, the dictionary $\mathbf{G} \in \mathbb{R}^{F\times K}$ is overcomplete with $K \geq F$, and the coefficients $\mathbf{p}_t \in \mathbb{R}^{K\times 1}$ are sparse. Both, $\mathbf{G}$ and $\{\mathbf{p}_t\}_{t=1}^T$ must be learned from the measurements $\{\mathbf{y}_t\}_{t=1}^T$ over T time slots. Several DL algorithms have been proposed in the literature, such as those in [24]–[27].

The DL framework has been used for RF cartography in the context of cognitive radio networks [16], [17], [28]. When used for modeling RSS, it is possible to interpret (2) as follows. For the RSS at frequency f and time t, $G_{fk} := [\mathbf{G}]_{fk}$ is the channel gain between the k-th transmitter and the receiver, while $[\mathbf{p}_t]_k$ is the transmit power at the k-th transmitter. Further, $\mathbf{p}_t$ is sparse since not all transmitters may be active at a given time t, and $K \geq F$ since the number of transmitters is unknown and presumably large. The transmitted power levels are dependent on the distance between the transmitter receiver pair. So, the power levels are considered to be randomly varying due to the impact of the distance from receiver and operating environment.

Taking the above interpretation further, modifications can be made to improve the process of learning $\mathbf{G}$. For instance, since channel gains and powers are always, positive, it is important to include the constraints $G_{fk} \geq 0$ and $\mathbf{p}_t \geq 0$. The following optimization problem was proposed in [28], and will serve as a starting point for the algorithms presented here.

$$\min_{\mathbf{G}\in\mathcal{G}, \{\mathbf{p}_t \geq 0\}} \frac{1}{2} \left\|\mathbf{Y} - \mathbf{G}\mathbf{P} - \mathbf{1}^T \otimes \boldsymbol{\nu}\right\|_F^2 + \lambda \sum_{t=1}^T \|\mathbf{p}_t\|_1 \tag{3}$$

where $\mathbf{Y} := [\mathbf{y}_1 \ \ldots \ \mathbf{y}_T]$, $\mathbf{P} := [\mathbf{p}_1 \ \ldots \ \mathbf{p}_T]$ and $\mathcal{G} := \{[\mathbf{g}_1 \ \ldots \ \mathbf{g}_K] \geq \mathbf{0} \mid \|\mathbf{g}_k\|_2 \leq 1, k = 1, \ldots, K\}$ and $\boldsymbol{\nu} \in \mathbb{R}_+^{F\times 1}$ accounts for non zero mean of noise. Note that a scaling ambiguity exists between the magnitude of entries in $\mathbf{G}$ and $\mathbf{p}$. In the present work, we resolve this ambiguity by making the columns of $\mathbf{G}$ unit norm, while allowing $\mathbf{p}$ to take arbitrary values. In practical settings, the values of $\mathbf{p}$ capture the path loss component of the channels that would otherwise be included within $\mathbf{G}$.

The least squares framework in (3) arises from the lack of prior knowledge about the noise characteristics. It is remarked that e_{ft} incorporates not only the receiver noise but also potentially unmodeled effects.

The optimization problem in (3) is non-convex since it involves the product of two variables $\mathbf{G}$ and $\mathbf{P}$. The problem is generally solved using block coordinate descent (BCD) method, which involves alternating minimization with respect to $\mathbf{G}$ and $\mathbf{P}$. Starting from an randomly initialized dictionary $\mathbf{G}_1$, the iteration t, the coefficient vector $\mathbf{p}_t$ is determined by solving

$$\hat{\mathbf{p}}_t = \underset{\mathbf{p}_t \geq \mathbf{0}}{\operatorname{argmin}} \frac{1}{2} \parallel \mathbf{y}_t - \boldsymbol{\nu} - \mathbf{G}\mathbf{p}_t \parallel_2^2 + \lambda \parallel \mathbf{p}_t \parallel_1 \tag{4}$$

where the ℓ_1 norm regularization is utilized to encourage sparsity in the coefficient vector. The regularization parameter λ generally depends on the environment, and must be tuned using the available data. Next, the dictionary is obtained by solving the following constrained quadratic problem

$$\mathbf{G} = \min_{\mathbf{G}\in\mathcal{G}} \frac{1}{2} \parallel \mathbf{Y} - \mathbf{G}\mathbf{P} - \mathbf{1}^T \otimes \boldsymbol{\nu} \parallel_F^2 . \tag{5}$$

Finally, the mean vector $\boldsymbol{\nu}$ is also obtained along the same lines.

The iterations continue till convergence, which occurs when the change in the values of $\mathbf{P}$ and $\mathbf{G}$ in successive iterations falls below a preset threshold. The full DL algorithm is summarized in Fig. 1.

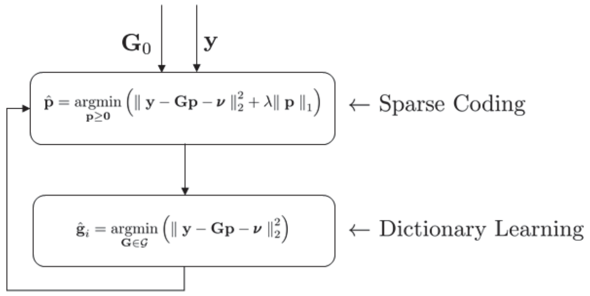

Fig. 1. Dictionary Learning Algorithm

III. DL BASED FINGERPRINTING

The fingerprint for each location is the dictionary learned from the RSS measurements. From the interpretation presented in Sec. II, the dictionary encodes the channel between the transmitters and the receiver. The entire process process of fingerprinting proceeds in two steps, namely training and mapping phases.

1) Training phase: Within the training phase, the receiver collects T vector observations at each location, and learns the corresponding dictionary. The dictionary learning algorithm used here is similar to that in [17], and is summarized in Algorithm 1. As discussed in Sec. II, the BCD algorithm is used to solve the problem in (3). Given an initial dictionary guess, often taken to be random, the sparse coefficient matrix P is learned and subsequently used to yield a better estimate of the dictionary. The dictionary is learned columnwise and projected on to the unit norm ball. These steps are repeated till convergence is achieved or the maximum number of iterations is reached. Convergence is achieved when the difference between the dictionary matrices at consecutive iterations falls below a preset threshold. **Remark:** The value of λ is data dependent, so it is chosen in the training phase which is then used in the mapping phase. To select the value of λ techniques like cross validation is used.

2) Mapping phase: Once the training phase is over for all locations of interest, the dictionaries are stored in a database. Given the RSS measurements from an unknown location, aim is to find the exact location using the stored dictionaries. In the mapping phase, first the sparse coding is performed for each dictionary and reconstruction error is calculated. The location corresponding to the dictionary with minimum reconstruction error is identified. The full algorithm is summarized in Algorithm 2.

It is remarked that the accuracy in the mapping phase depends on the number of samples used. Using one or two samples is generally not enough, since the transmitter activity over a short duration of time may simply resemble that at another location. Best performance is obtained if five or more samples of RSS measurements are used in mapping phase.

As in previous section estimation using DL is completely blind. However if prior information about channel distribution

Algorithm 1 Dl based fingerprinting (Training)

Input:=Y, initial dictionary **G**, λ
Output:= $\hat{\mathbf{P}}$, $\hat{\mathbf{G}}$
Algo parameters: **A=B=0**, **c=d=**$\boldsymbol{\nu}$**=0**.
repeat
    Sparse Coding Step
    for t=1:T **do**
        solve for $\mathbf{P}_t$ using (4)
    end for
    Compute: $\mathbf{A} = \hat{\mathbf{P}}\hat{\mathbf{P}}^T$, $\mathbf{B} = \mathbf{Y}\hat{\mathbf{P}}^T$, $\mathbf{c} = \mathbf{Y1}$, $\mathbf{d} = \hat{\mathbf{P}}\mathbf{1}$
    repeat
        $\tilde{\mathbf{B}} = \mathbf{B} - \boldsymbol{\nu}\mathbf{d}^T$
        for k=1,..,K **do**
            $\tilde{\mathbf{g}}_k = \frac{1}{\mathbf{A}(k,k)} \left(\tilde{\mathbf{b}}_k - \mathbf{G}\mathbf{a}_k \right) + \mathbf{g}_k$
            $\hat{\mathbf{g}}_k = \frac{[\tilde{\mathbf{g}}_k]^+}{\max\{\|[\tilde{\mathbf{g}}_k]^+\|_2, 1\}}$
        end for
        $\boldsymbol{\nu} = \left[\mathbf{c} - \hat{\mathbf{G}}\mathbf{d} \right]^+$
    until Convergence
until Convergence

Algorithm 2 Mapping Phase

Input: $\mathbf{Y}_{test}$, learned dictionaries $\hat{\mathbf{G}}_1, \ldots, \hat{\mathbf{G}}_M$ for M locations, T_m, λ
Output: location $\hat{i}$
Sparse Coding Step
for $i = 1$ to M **do**
    for $t = 1$ to T_m **do**
        solve for $\{\hat{\mathbf{p}}_t^i\}$ using (4)
    end for
    Calculate for each receiver
$\eta_i = \sum_{t=1}^{T_m} \left\| \mathbf{y}_t - \hat{\mathbf{G}}_i \hat{\mathbf{p}}_t^i \right\|_2^2$
end for
Return $\hat{i} = \arg\min \{\eta_1, \ldots, \eta_M\}$

is available, the DL algorithm can be appropriately modified. To this end, the optimization step in (3) can be written as

$$\mathbf{G} = \operatorname{argmax} \; l(\mathbf{G}|\mathbf{Y}) \tag{6}$$

where $l(\mathbf{G}|\mathbf{Y})$ is the log likelihood function and is given as

$$l(\mathbf{G}|\mathbf{Y}) = \log \; f(\mathbf{G}|\mathbf{Y}) \tag{7}$$

$$f(\mathbf{G}|\mathbf{Y}) = f(\mathbf{Y}|\mathbf{G}) f(\mathbf{G}) \tag{8}$$

$$l(\mathbf{G}|\mathbf{Y}) = \log \; f(\mathbf{Y}|\mathbf{G}) + \log \; f(\mathbf{G}) \tag{9}$$

$$\hat{\mathbf{G}} = \operatorname{argmin} \; -l(\mathbf{G}|\mathbf{Y}) \tag{10}$$

$$\hat{\mathbf{G}} = \operatorname{argmin} \; -(\log \; f(\mathbf{Y}|\mathbf{G}) + \log \; f(\mathbf{G})) \tag{11}$$

From (11), we observe that, incorporating prior information results in addition of extra regularization term.

3) Algorithms for DL assuming prior information: For a given channel distribution, channel gains can be obtained as square of channel coefficients. Here we consider, channel to be Rayleigh distributed therefore, channel gain comes out to be exponentially distributed. Let the probability density function of each column $\mathbf{g}_i$ be given by

$$f(\mathbf{g}_i) = \gamma e^{-\gamma \sum\limits_{j=1}^{F} g_{ij}}. \tag{12}$$

Therefore, the log likelihood of $f(\mathbf{g}_i)$ is given by

$$\log f(\mathbf{g}_i) = -\gamma - \gamma \sum_{j=1}^{F} g_{ij}. \tag{13}$$

Ignoring constant terms, the form of the log-likelihood function suggests that overall optimization problem for DL should be posed as

$$\min_{\mathbf{G}\in\mathcal{G},\mathbf{P}\geq\mathbf{0},\boldsymbol{\nu}\geq\mathbf{0}} \frac{1}{2} \parallel \mathbf{Y} - \mathbf{GP} - \mathbf{1}^T \otimes \boldsymbol{\nu} \parallel_F^2 + \lambda \sum_{t=1}^{T} \|\mathbf{p}_t\|_1 + \gamma \sum_i \sum_j G_{ij} \tag{14}$$

In this section we present the algorithms for solving (14) in batch and online fashion.

- **Batch Implementation:** The objective function for the batch algorithm is given by equation (14). As before, $\mathbf{G}$ is estimated columnwise, which results in objective function given by

$$\hat{\mathbf{g}}_i = \min_{\mathbf{g}_i\in\mathcal{G}} \frac{1}{2} \parallel \mathbf{Y} - \mathbf{GP} - \mathbf{1}^T \otimes \boldsymbol{\nu} \parallel_F^2 + \gamma \sum_{j=1}^{F} g_{ij} \tag{15}$$

Batch implementation with prior information is similar to one provided by Algorithm 1, except that step 13 must be replaced with

$$\tilde{\mathbf{g}}_k = \frac{1}{\mathbf{A}(k,k)} \left(\tilde{\mathbf{b}}_k - \mathbf{Ga}_k - \gamma\mathbf{1}\right) + \mathbf{g}_k \tag{16}$$

- **Online Implementation** The batch algorithm requires the observation matrix to be stored before any processing. This can be tricky while handling the large amount of data. So, the online algorithm is desirable so that the raw measurements can be discarded over time. It uses intermediate calculations which do not require storage of previous observations. Another advantage of online implementation is the learning process is more efficient as these can well adapt to dynamic environments. Further, the batch algorithm is also more computationally complex, and the online algorithm yields near optimal dictionaries if the number of samples collected is sufficiently large. So instead of learning the features in batch training it is advocated to use the online algorithm. To this end, the optimization problem is modified as follows:

$$\min_{\mathbf{G}\in\mathcal{G},\mathbf{P}\geq\mathbf{0},\boldsymbol{\nu}\geq\mathbf{0}} \sum_{t=1}^{T} \frac{1}{2} \parallel \mathbf{y}_t - \mathbf{Gp}_t - \boldsymbol{\nu} \parallel_2^2 + \lambda \parallel \mathbf{p}_t \parallel_1 + \gamma \sum_i \sum_j G_{ij} \tag{17}$$

The full online algorithm is summarized in Algorithm 3. The training data i.e. RSS measurements given at a time instant are used to generate the sparse vector followed by the dictionary modification. The learning process in Algorithm 3 is different from Algorithm 1 as it does not require multiple convergence loops and intermediate calculations require vector manipulations instead of matrix.

Algorithm 3 Estimation of P and G using Exponential Prior (online Implementation)

Input:=$\mathbf{y}_t$, initial dictionary $\mathbf{G}(0)$, λ, and γ
Output:=$\hat{\mathbf{p}}_t$, $\hat{\mathbf{G}}_t$
Algo parameters: $\mathbf{A}(0)$=$\mathbf{B}(0)$=$\mathbf{0}$, $\mathbf{c}(0)$=$\mathbf{d}(0)$=$\mathbf{j}(0)$=$\boldsymbol{\nu}(0)$=$\mathbf{0}$.
for t=1:T **do**
    Sparse Coding Step
    $\hat{\mathbf{p}}_t = \underset{\mathbf{p}\geq\mathbf{0}}{\operatorname{argmin}} \frac{1}{2} \parallel \mathbf{y}_t - \boldsymbol{\nu}_{t-1} - \mathbf{G}_{t-1}\mathbf{p}_t \parallel_2^2 + \lambda \parallel \mathbf{p}_t \parallel_1$
    Dictionary Learning Step
    Compute:
    $\mathbf{A}_t = \mathbf{A}_{t-1} + \hat{\mathbf{p}}_t\hat{\mathbf{p}}_t^T$,
    $\mathbf{B}_t = \mathbf{B}_{t-1} + \mathbf{y}_t\hat{\mathbf{p}}_t^T$,
    $\mathbf{c}_t = \mathbf{c}_{t-1} + \mathbf{y}_t$,
    $\mathbf{d}_t = \mathbf{d}_{t-1} + \hat{\mathbf{p}}_t$
    $\mathbf{j}_t = \mathbf{j}_{t-1} + \gamma\mathbf{1}$
    repeat
        $\tilde{\mathbf{B}}_t = \mathbf{B}_t - \boldsymbol{\nu}_t\mathbf{d}_t^T$
        for k=1,..,K **do**
            $\tilde{\mathbf{g}}_k = \frac{1}{\mathbf{A}(k,k)} \left(\tilde{\mathbf{b}}_k - \mathbf{Ga}_k - \mathbf{j}_t\right) + \mathbf{g}_k$
            $\hat{\mathbf{g}}_k = \frac{[\tilde{\mathbf{g}}_k]^+}{\max\{\|[\tilde{\mathbf{g}}_k]^+\|_2, 1\}}$
        end for
        $\boldsymbol{\nu}_t = \left[\mathbf{c}_t - \hat{\mathbf{G}}\mathbf{d}_t\right]^+$
    until Convergence
end for

IV. Simulation Results

This section provides simulation results that demonstrate the effectiveness of the proposed DL-based fingerprinting technique. Existing fingerprinting techniques that utilize the path loss model cannot cope with time-varying transmitter activity, and are therefore not compared here. Within the context of blind fingerprinting of time-varying RSS, another possibility is that of using classification algorithms that can learn to discriminate between locations. The performance of the proposed algorithm is compared with standard classification algorithms.

The simulation setup considers a total of ten transmitters, spread over a unit area as shown Fig. 2, and transmitting intermittently over five frequencies. The transmit powers are assumed to be uniformly distributed between 0.9 and 1, and the transmitters are active with probability 0.3. The figure also shows the two receiver locations, and the channel between a transmitter-receiver pair is modeled as $\left(\frac{d_{mk}}{d_0}\right)^{-\alpha} |h_{fk}|^2$ where $d_0 = 0.1$, $\alpha = 2$, d is the distance between the transmitter and the receiver, and $|h_{fk}|^2$ is the exponentially distributed random variable modeling the small scale fading. The training phase

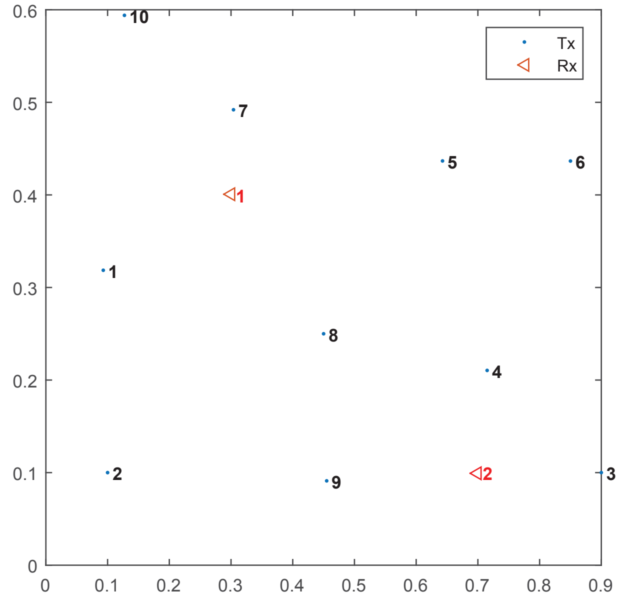

Fig. 2. Transmitter and Receiver Locations

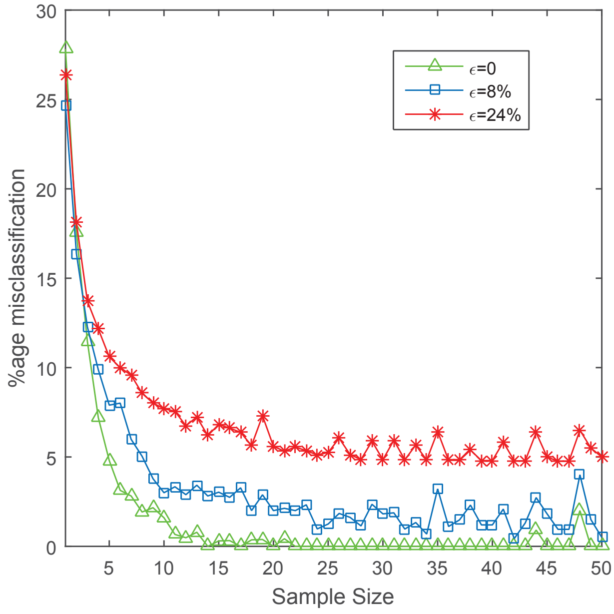

Fig. 4. Percentage misclassification vs sample size for DL

acquires 1000 samples to learn the overcomplete dictionary of size 5×15. The subproblem in (4) is solved using interior point method [29]. Finally, although λ must be chosen via cross-validation, the algorithm was found to work well for the range $[0.1, 0.9]$. For the present set of simulations, we set $\lambda = 0.5$.

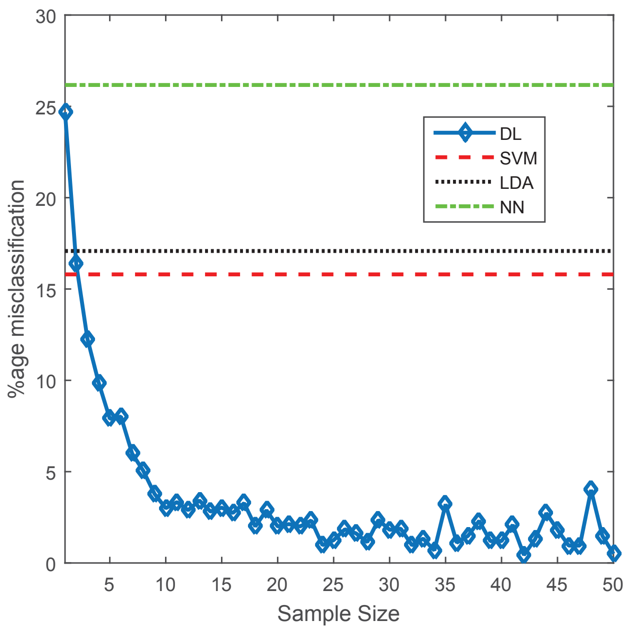

Fig. 3. Percentage misclassification for four different algorithms

To begin with, Fig. 3 shows the misclassification error incurred by the proposed error, calculated as the percentage of time the algorithm incorrectly identified each receiver location. The misclassification error incurred by proposed method is compared with classical classification algorithms, such as support vector machines(SVM), linear discriminant analysis (LDA), and nearest-neighbor classifier (NN). The performance of the proposed DL-based fingerprinting algorithm is better than the standard classifiers, especially if more than 3 samples are available during the mapping phase. Note that, in theory, it is possible to consider improved classifiers that utilize prior information for differentiating between different locations. However, construction of such multi-class discriminants would still require batch processing of samples from all possible receiver locations. In contrast, the proposed algorithm generates fingerprints for each set of samples separately, and is therefore scalable to any number of receiver locations.

The performance of the DL-based algorithm is expected to suffer in the presence of noisy RSS readings. Fig. 4 plots the misclassification error with the number of samples. The different curves are plotted for different ratios of standard deviation of the RSS error to the average channel gain, expressed as a percentage. As expected, the misclassification error is high if the RSS measurements are too noisy, even for large number of samples.

Next, the performance of the online DL algorithm is evaluated, and the effect of prior information is analyzed. To this end, the reconstruction error in the training phase, given by $\eta = \mathbb{E}\left[\left\|\mathbf{y}_t - \hat{\mathbf{G}}\hat{\mathbf{p}}_t - \hat{\boldsymbol{\nu}}\right\|\right]$, is plotted with time in Fig. 5. The reconstruction error improves with time, as the dictionary becomes more representative of the RSS measurements at a particular location. The use of prior information speeds up the convergence of the algorithm, allowing shorter training times. It is remarked that the online algorithm requires further tuning

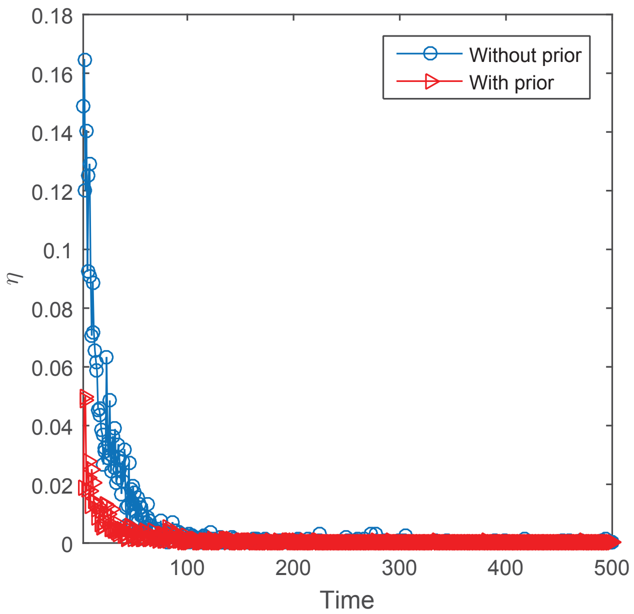

Fig. 5. Training reconstruction error with time for the online fingerprinting method

parameters to run correctly. In the present case, $\lambda = 10^{-4}$, tuned from measurements.

V. Conclusion

This paper considers the problem of radio frequency fingerprinting for localization and mapping in indoor environments. To this end, dictionary learning (DL) has been proposed for fingerprinting, based on the received signal strength at different frequencies. Preliminary results show that DL is able to classify different locations correctly, regardless of their time-varying activity and random fading impairments. Prior information regarding channel distribution has also been included to improve the DL performance and an online algorithm is proposed for allowing low complexity implementation. Future research includes the analysis of spatial resolution afforded by the proposed scheme, as well as implementation over a testbed.

References

[1] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Communications Surveys & Tutorials*, pp. 13–32, 2009.

[2] A. Khalajmehrabadi, N. Gatsis, and D. Akopian, "Modern wlan fingerprinting indoor positioning methods and deployment challenges," *IEEE Communications Surveys & Tutorials*, 2017.

[3] R. Mautz, "Combination of indoor and outdoor positioning," in *1st International Conference on Machine Control & Guidance*, 2008.

[4] S. Chan and G. Sohn, "Indoor localization using wi-fi based fingerprinting and trilateration techiques for lbs applications," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 38, p. 4, 2012.

[5] M. Azizyan, I. Constandache, and R. Roy Choudhury, "Surroundsense: mobile phone localization via ambience fingerprinting," in *Proceedings of the 15th annual international conference on Mobile computing and networking*, 2009, pp. 261–272.

[6] A. Yves and P. Hao, "Rssi-based indoor localization using rssi-with-angle-based localization estimation algorithm," *Sensor Netw Data Commun*, vol. 4, no. 122, p. 2, 2015.

[7] M. Shchekotov, "Indoor localization method based on wi-fi trilateration technique," in *Proceeding of the 16th conference of fruct association*, 2014, pp. 177–179.

[8] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: wireless indoor localization with little human intervention," in *Proceedings of the 18th annual international conference on Mobile computing and networking*, 2012.

[9] J. Huang, D. Millman, M. Quigley, D. Stavens, S. Thrun, and A. Aggarwal, "Efficient, generalized indoor wifi graphslam," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 1038–1043.

[10] M. Bshara, U. Orguner, F. Gustafsson, and L. Van Biesen, "Fingerprinting localization in wireless networks based on received-signal-strength measurements: A case study on wimax networks," *IEEE Transactions on Vehicular Technology*, pp. 283–294, 2010.

[11] Y.-S. Lu, C.-F. Lai, C.-C. Hu, Y.-M. Huang, and X.-H. Ge, "Path loss exponent estimation for indoor wireless sensor positioning." *KSII Transactions on Internet & Information Systems*, 2010.

[12] F. Ebner, T. Fetzer, F. Deinzer, and M. Grzegorzek, "On wi-fi model optimizations for smartphone-based indoor localization," *ISPRS International Journal of Geo-Information*, 2017.

[13] I. Tošić and P. Frossard, "Dictionary learning," *IEEE Signal Processing Magazine*, pp. 27–38, 2011.

[14] I. Rish and G. Grabarnik, *Sparse Modeling: Theory, Algorithms, and Applications*. CRC Press, 2014.

[15] G. Mateos and K. Rajawat, "Dynamic network cartography: Advances in network health monitoring," *IEEE Signal Processing Magazine*, pp. 129–143, 2013.

[16] S.-J. Kim and G. Giannakis, "Cognitive radio spectrum prediction using dictionary learning," in *IEEE Global Communications Conference (GLOBECOM), 2013*. IEEE, 2013, pp. 3206–3211.

[17] W. Ke and L. Wu, "Indoor localization in the presence of rss variations via sparse solution finding and dictionary learning," *Progress In Electromagnetics Research*, pp. 353–368, 2012.

[18] B.-Q. Yang, C.-C. Gu, K.-J. Wu, T. Zhang, and X.-P. Guan, "Simultaneous dimensionality reduction and dictionary learning for sparse representation based classification," *Multimedia Tools and Applications*, pp. 8969–8990, 2017.

[19] S. Kong and D. Wang, "A dictionary learning approach for classification: separating the particularity and the commonality," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 186–199.

[20] K. Shi, Z. Ma, R. Zhang, W. Hu, and H. Chen, "Support vector regression based indoor location in ieee 802.11 environments," *Mobile Information Systems*, 2015.

[21] Y. Xie, Y. Wang, A. Nallanathan, and L. Wang, "An improved k-nearest-neighbor indoor localization method based on spearman distance," *IEEE Signal Processing Letters*, pp. 351–355, 2016.

[22] M. S. Ifthekhar, N. Saha, and Y. M. Jang, "Neural network based indoor positioning technique in optical camera communication system," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2014, pp. 431–435.

[23] K. Engan, S. O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 1999*. IEEE, 1999, pp. 2443–2446.

[24] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, pp. 4311–4322, 2006.

[25] K. Skretting and K. Engan, "Recursive least squares dictionary learning algorithm," *IEEE Transactions on Signal Processing*, pp. 2121–2130, 2010.

[26] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 689–696.

[27] S.-J. Kim, N. Jain, G. B. Giannakis, P. Forero *et al.*, "Joint link learning and cognitive radio sensing," in *Asilomar Conference on Signals, Systems and Computers (ASILOMAR), 2011*. IEEE, 2011, pp. 1415–1419.

[28] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale l_1-regularized least squares," *IEEE Journal of Selected Topics in Signal Processing*, pp. 606–617, 2007.