HAL
open science

Ridge regression and Kalman filtering for target tracking in wireless sensor networks

Sandy Mahfouz, Farah Mourad-Chehade, Paul Honeine, Joumana Farah, Hichem Snoussi

To cite this version:

Sandy Mahfouz, Farah Mourad-Chehade, Paul Honeine, Joumana Farah, Hichem Snoussi. Ridge regression and Kalman filtering for target tracking in wireless sensor networks. Proc. eighth IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM), 2014, A Coruna, Spain. pp.237-240, 10.1109/SAM.2014.6882384 . hal-01965991

HAL Id: hal-01965991

https://hal.science/hal-01965991

Submitted on 27 Dec 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

RIDGE REGRESSION AND KALMAN FILTERING FOR TARGET TRACKING IN WIRELESS SENSOR NETWORKS

Sandy Mahfouz, Farah Mourad-Chehade*, Paul Honeine*, Joumana Farah†, Hichem Snoussi**

*Institut Charles Delaunay (CNRS), Université de Technologie de Troyes, Troyes, France

†Department of telecommunications, Faculty of engineering, Université Saint-Esprit de Kaslik, Lebanon

ABSTRACT

This paper introduces an original method for target tracking in wireless sensor networks that combines machine learning and Kalman filtering. A database of radio-fingerprints is used, along with the ridge regression learning method, to compute a model that takes as input RSSI information, and yields, as output, the positions where the RSSIs are measured. This model leads to a position estimate for each target. The Kalman filter is used afterwards to combine the model's estimates with predictions of the target's positions based on acceleration information, leading to more accurate ones.

***Index Terms*—** radio-fingerprinting, Kalman filter, ridge regression, RSSI, tracking, WSN.

I. INTRODUCTION

Recently, wireless sensor networks (WSNs) have become a major research field, since they can be used for many everyday applications [1], [2]. One interesting application of WSNs is target tracking [3], that consists of estimating instantly the position of a moving target. Target tracking can be viewed as a sequential location estimation problem, that requires a real-time location estimation algorithm. Typically, sensors broadcast signals in the network, while targets collect these signals for location estimation. Several types of measurements can be considered, such as received signal strength indicators (RSSIs) [4], that has proven to achieve acceptable performance, with no extra hardware.

Many RSSI-based tracking algorithms have been proposed in the literature. For instance, the authors of [5] provide a tracking technique using a particle filter with the exact RSSI channel model. Using also signals powers, the authors of [6] propose a target tracking method based on the interval theory and connectivity measurements. These methods are more robust than the ones using the exact channel model. However, the number and the positions of the sensors in the network highly affect the performance of such methods.

In other contexts, RSSI-based methods have been proposed for nodes localization in WSNs. Such methods investigate observation information, but do not take advantage of nodes mobility. Radio-fingerprinting [7] is an interesting RSSI-based localization approach, since it avoids estimating the RSSI channel model. We have recently proposed in [8] and [9] two robust localization methods using radio-fingerprinting in WSNs, by taking advantage of kernel methods in machine learning. In [10], the authors perform tracking by combining estimates obtained using the weighted K-nearest neighbor (WKNN) algorithm, which is a well-known RSSI-based fingerprinting algorithm, with acceleration information using the Kalman filter with a second-order state-space model.

In this paper, we provide a new method for target tracking in WSNs by combining radio-fingerprinting and accelerometer information. Our method consists of constructing a database of radio-fingerprints used with the ridge regression learning method [11] to define a kernel-based model, whose input is the RSSI vector and whose output is the corresponding position. Then, RSSI measures collected by a moving target are used with the computed kernel-based model to get a position estimate. This estimate is combined with the acceleration information, by means of a Kalman filter [12], to achieve better accuracy. The rest of the paper is organized as follows. The proposed tracking method is described in Section II. Section III describes the kernel-based model's definition, while Section IV shows how the Kalman filter is applied to make use of the accelerometer information. In Section V, we examine the performance of the proposed method. Finally, Section VI concludes the paper.

II. DESCRIPTION OF THE TRACKING METHOD

Consider an environment of D dimensions, with $D = 2$ or 3, and N_s stationary sensors having known locations, denoted by $\boldsymbol{s}_i,\ i \in \{1, \ldots, N_s\}$. In the following, all coordinates are 1-by-D vectors. For the sake of clarity, only one target with the unknown position $\boldsymbol{x}(k)$ is considered in this paper, k being the current time step.

The first step in the proposed method consists of setting reference positions along the network where RSSI measurements are collected, to construct a database of radio-fingerprints. To this end, N_p reference positions, denoted by $\boldsymbol{p}_\ell$, $\ell \in \{1, \ldots, N_p\}$, are generated uniformly or randomly in the studied region. The stationary sensors broadcast signals

This work is supported by the Champagne-Ardenne Region in France.

with the same initial power, and signals are assumed to be received at all reference positions. A sensor is temporarily placed at each reference position $\boldsymbol{p}_\ell$ to measure the RSSI of the broadcasted signals. Let $\boldsymbol{\rho}_\ell = (\rho_{\boldsymbol{s}_1,\boldsymbol{p}_\ell} \dots \rho_{\boldsymbol{s}_{N_s},\boldsymbol{p}_\ell})^\top$ be the vector of RSSIs sent by all N_s sensors and received at the position $\boldsymbol{p}_\ell$. This way, a database of N_p pairs $(\boldsymbol{\rho}_\ell, \boldsymbol{p}_\ell)$ is obtained, where $\ell \in \{1, \dots, N_p\}$. The objective now is to define a function $\boldsymbol{\psi}\colon \mathbb{R}^{N_s} \mapsto \mathbb{R}^D$, that associates to each RSSI vector $\boldsymbol{\rho}_\ell$ the corresponding position $\boldsymbol{p}_\ell$. Kernel methods [13] provide an elegant framework to find $\boldsymbol{\psi}(\cdot)$, as it will be shown in the following section.

Note that the database construction and the computation of $\boldsymbol{\psi}(\cdot)$ are performed only once, before the tracking phase. Once $\boldsymbol{\psi}(\cdot)$ is defined, the target is able to perform all tracking computations and determine its own position. Indeed, consider a target in the network. This target collects, while moving, the RSSIs of the signals received from the N_s sensors, at each time step k, and stores them into a vector $\boldsymbol{\rho}(k)$. An estimate of the target's position is obtained using the defined model $\boldsymbol{\psi}(\cdot)$, and the estimated coordinates are then given by:

$$\boldsymbol{z}(k) = \boldsymbol{\psi}(\boldsymbol{\rho}(k)). \tag{1}$$

Each target is assumed to be equipped with an accelerometer that yields its instantaneous accelerations over the D coordinates. The estimated position $\boldsymbol{z}(k)$ is then combined with the accelerometer information to obtain a more accurate estimate using the Kalman filter [12], as will be explained in Section IV.

III. COMPUTATION OF $\boldsymbol{\psi}(\cdot)$

Based on the information gathered by the target from the stationary sensors (i.e., the constructed database), the objective is to define the model $\boldsymbol{\psi}(\cdot)$, that associates to each RSSI vector $\boldsymbol{\rho}_\ell$ the corresponding position $\boldsymbol{p}_\ell$, $\ell \in \{1, \dots, N_p\}$. Determining $\boldsymbol{\psi}(\cdot)$ requires solving a nonlinear regression problem, as we show in the following.

Let $\boldsymbol{\psi}(\cdot) = (\psi_1(\cdot) \dots \psi_D(\cdot))$, where each $\psi_d(\cdot)$, $d \in \{1, \dots, D\}$, estimates the d-th coordinate in $\boldsymbol{p}_\ell = (p_{\ell,1} \dots p_{\ell,D})$, for an input $\boldsymbol{\rho}_\ell$. Let $\boldsymbol{P} = (\boldsymbol{p}_1^\top \dots \boldsymbol{p}_{N_p}^\top)^\top$; the matrix $\boldsymbol{P}$ is then of size N_p-by-D, having $p_{\ell,d}$ for the (ℓ, d)-th entry. In the following, we denote $\boldsymbol{p}_\ell$ by $\boldsymbol{P}_{\ell,*}$, and the d-th column of $\boldsymbol{P}$ by $\boldsymbol{P}_{*,d}$. The vector $\boldsymbol{P}_{*,d}$ holds now all N_p points for a fixed coordinate d. The classic ridge regression is considered in this section, where D optimization problems are set separately to estimate the D models $\psi_d(\cdot)$ [11]. Indeed, each function $\psi_d(\cdot)$ must minimize the mean quadratic error between the model's outputs $\psi_d(\boldsymbol{\rho}_\ell)$ and the desired outputs $p_{\ell,d}$:

$$\min_{\psi_d \in \mathcal{H}} \frac{1}{N_p} \sum_{\ell=1}^{N_p} (p_{\ell,d} - \psi_d(\boldsymbol{\rho}_\ell))^2 + \eta \|\psi_d\|_{\mathcal{H}}^2, \tag{2}$$

where η is a regularization parameter that controls the tradeoff between the training error and the complexity of the solution. According to the representer theorem [13], the optimal function can be written as follows:

$$\psi_d(\cdot) = \sum_{\ell=1}^{N_p} \alpha_{\ell,d}\, \kappa(\boldsymbol{\rho}_\ell, \cdot), \tag{3}$$

where $\kappa : \mathbb{R}^{N_s} \times \mathbb{R}^{N_s} \mapsto \mathbb{R}$ is a reproducing kernel, and $\alpha_{\ell,d}, \ell \in \{1, \dots, N_p\}$, are parameters to be determined. Let us denote by $\boldsymbol{\alpha}$ the N_p-by-D matrix whose (ℓ, d)-th entry is $\alpha_{\ell,d}$. Let then $\boldsymbol{\alpha}_{*,d}$ denote its d-th column, and $\boldsymbol{\alpha}_{\ell,*}$ its ℓ-th row. By injecting (3) in (2), we get a dual optimization problem in terms of $\boldsymbol{\alpha}_{*,d}$, whose solution is given by taking its derivative with respect to $\boldsymbol{\alpha}_{*,d}$ and setting it to zero. One can easily find the following form of the solution:

$$\boldsymbol{\alpha}_{*,d} = (\boldsymbol{K} + \eta N_p \boldsymbol{I}_{N_p})^{-1} \boldsymbol{P}_{*,d},$$

where $\boldsymbol{I}_{N_p}$ is the N_p-by-N_p identity matrix, and $\boldsymbol{K}$ is the N_p-by-N_p matrix whose (i, j)-th entry is $\kappa(\boldsymbol{\rho}_i, \boldsymbol{\rho}_j)$, for $i, j \in \{1, ..., N_p\}$. For an appropriate value of the regularization parameter η, the matrix between parenthesis is always non-singular.

It is easy to see that the same matrix $(\boldsymbol{K} + \eta N_p \boldsymbol{I}_{N_p})$ needs to be inverted in order to estimate each coordinate. Therefore, to reduce the computational complexity, all D estimations are collected into a single matrix inversion problem, as follows:

$$\boldsymbol{\alpha} = (\boldsymbol{K} + \eta N_p \boldsymbol{I}_{N_p})^{-1} \boldsymbol{P}. \tag{4}$$

We then define a model that allows us to estimate all D coordinates at once, using equation (3) and the definition of the vector of functions $\boldsymbol{\psi}(\cdot)$, as follows:

$$\boldsymbol{\psi}(\cdot) = \sum_{\ell=1}^{N_p} \boldsymbol{\alpha}_{\ell,*}\, \kappa(\boldsymbol{\rho}_\ell, \cdot). \tag{5}$$

IV. TRACKING ENHANCEMENT USING THE KALMAN FILTER

In this section, we propose to combine the position estimates obtained using the already-defined kernel-based model with accelerometer information by means of a Kalman filter [12]. Challenges arise when the target does not follow a predictable path, making it difficult to find a model to describe its motion. We propose here to use a third-order state-space model to describe the target's motion. Such model considers that the target's accelerations vary linearly between any two consecutive time steps $k-1$ and k with a slope equal to $\frac{\boldsymbol{\gamma}(k) - \boldsymbol{\gamma}(k-1)}{\Delta t}$, where $\boldsymbol{\gamma}(k) = (\gamma_1(k) \dots \gamma_D(k))$ is the acceleration vector at time step k, and Δt the time period separating two consecutive time steps. According to this assumption, the velocity vector $\boldsymbol{\nu}(k)$ of the target at time step k is estimated recursively by:

$$\boldsymbol{\nu}(k) = \boldsymbol{\nu}(k-1) + \boldsymbol{\gamma}(k-1)\, \Delta t + \frac{\boldsymbol{\gamma}(k) - \boldsymbol{\gamma}(k-1)}{\Delta t} \frac{\Delta t^2}{2}. \tag{6}$$

Then, by taking the primitive integral of (6), the position of the target can be written as follows:

$$\boldsymbol{x}(k) = \boldsymbol{x}(k-1) + \boldsymbol{\nu}(k-1)\,\Delta t \,+\, \boldsymbol{\gamma}(k-1)\,\frac{\Delta t^2}{2} + \frac{\boldsymbol{\gamma}(k)-\boldsymbol{\gamma}(k-1)}{\Delta t}\,\frac{\Delta t^3}{6}. \quad (7)$$

For small values of Δt, one can find an accurate approximation of $\boldsymbol{x}(k)$ using this model. The target is assumed to be at a fixed known position $\boldsymbol{x}(0)$ at the beginning of the tracking, having $\boldsymbol{\nu}(0)$ and $\boldsymbol{\gamma}(0)$ null. We add a random D-vector noise $\boldsymbol{\theta}(k)$ to the model of (7), to take into account the inaccuracies in the measured accelerations. Let the acceleration noises be independent with zero-mean normal distributions, having known variances $\sigma^2_{\gamma_d}$, $d = 1, ..., D$. Then, $\boldsymbol{\theta}(k)$ would have a normal probability distribution, with zero mean and diagonal covariance matrix $\boldsymbol{Q}(k)$, given by the following:

$$\boldsymbol{Q}(k) = \boldsymbol{Q}(k-1) + \boldsymbol{Q}_{\boldsymbol{\nu}}(k-1)\,\Delta t^2 + \frac{11}{36}\Delta t^4\,\mathrm{Diag}\left(\boldsymbol{\sigma}^2_{\boldsymbol{\gamma}}\right),$$

where $\boldsymbol{Q}(0)$ is null, since there is no uncertainty at time step $k = 0$, and $\boldsymbol{Q}_{\boldsymbol{\nu}}(k)$ is given by:

$$\boldsymbol{Q}_{\boldsymbol{\nu}}(k) = \boldsymbol{Q}_{\boldsymbol{\nu}}(k-1) + \frac{3}{2}\Delta t^2\,\mathrm{Diag}\left(\boldsymbol{\sigma}^2_{\boldsymbol{\gamma}}\right),$$

where $\boldsymbol{Q}_{\boldsymbol{\nu}}(0)$ is also null, and $\mathrm{Diag}\left(\boldsymbol{\sigma}^2_{\boldsymbol{\gamma}}\right)$ is the D-by-D diagonal matrix with entries $\sigma^2_{\gamma_d}$, $d = 1, ..., D$.

From (7), one can see that the relation between $\boldsymbol{x}(k)$ and $\boldsymbol{x}(k-1)$ is linear, and can be written as follows:

$$\boldsymbol{x}(k) = \boldsymbol{x}(k-1) + \boldsymbol{B}(k) + \boldsymbol{\theta}(k), \quad (8)$$

where $\boldsymbol{B}(k)$ is a control-input vector depending on the accelerations given by $\boldsymbol{\nu}(k-1)\,\Delta t + \boldsymbol{\gamma}(k-1)\,\frac{\Delta t^2}{2} + \frac{\boldsymbol{\gamma}(k)-\boldsymbol{\gamma}(k-1)}{\Delta t}\,\frac{\Delta t^3}{6}$. Now, to consider the observations, instead of taking the RSSI values, we take the position estimate $\boldsymbol{z}(k)$ given by the kernel-based model, to which we add an additive noise $\boldsymbol{n}(k)$ due to measurements, as follows:

$$\boldsymbol{z}(k) = \boldsymbol{x}(k) + \boldsymbol{n}(k). \quad (9)$$

The noise $\boldsymbol{n}(k) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R})$ is assumed to have a normal distribution, with zero mean and covariance matrix $\boldsymbol{R}$. To approximate $\boldsymbol{R}$, a new set of reference pairs is generated, and the positions are estimated using $\boldsymbol{\psi}(\cdot)$. The error on this set is computed and stored into a vector. The matrix $\boldsymbol{R}$, assumed to be constant over time and for all targets, is then determined by computing the covariance of the error vector.

Finally, having the two linear equations (8) and (9), the Kalman filter can be applied to enhance the solution provided by the technique of Section III. Indeed, the Kalman filter first predicts the unknown position using the previous estimated position and the state-space equation (8), as follows:

$$\hat{\boldsymbol{x}}^-(k) = \hat{\boldsymbol{x}}(k-1) + \boldsymbol{B}(k),$$

where $\hat{\boldsymbol{x}}(k-1)$ denotes the target's position estimated by the Kalman filter at time step $k-1$, with $\hat{\boldsymbol{x}}(0)$ assumed to be known. The D-by-D predicted estimation covariance is also updated by $\boldsymbol{T}^-(k) = \boldsymbol{T}(k-1) + \boldsymbol{Q}(k)$, where $\boldsymbol{T}(k-1)$ is the final covariance estimation at time step $k-1$, and $\boldsymbol{T}(0)$ is null since the initial state is known. Finally, the predicted quantities $\hat{\boldsymbol{x}}^-(k)$ and $\boldsymbol{T}^-(k)$ are corrected using the observation equation (9) as follows:

$$\hat{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}^-(k) + \left(\boldsymbol{z}(k) - \hat{\boldsymbol{x}}^-(k)\right)\boldsymbol{G}_K(k)$$
$$\boldsymbol{T}(k) = \left(\boldsymbol{I}_D - \boldsymbol{G}_K(k)\right)\boldsymbol{T}^-(k),$$

where $\boldsymbol{I}_D$ is the D-by-D identity matrix, and $\boldsymbol{G}_K(k)$ is the optimal Kalman gain given by:

$$\boldsymbol{G}_K(k) = \boldsymbol{T}^-(k)\left(\boldsymbol{T}^-(k) + \boldsymbol{R}\right)^{-1}.$$

In summary, a target's position is obtained by combining the position estimated using the model $\boldsymbol{\psi}(\cdot)$ with the one predicted by the accelerometer information.

V. PRACTICAL SIMULATIONS AND RESULTS

In this section, we evaluate the performance of our method on simulated data. We consider a $100m \times 100m$ area, and generate 16 stationary sensors and 100 reference positions uniformly distributed over the area. RSSI values are computed using the known Okumura-Hata model [14] given by:

$$\rho_{\boldsymbol{s}_i,\boldsymbol{p}_\ell} = \rho_0 - 10\,n_P\,\log_{10}\|\boldsymbol{s}_i - \boldsymbol{p}_\ell\| + \varepsilon_{i,\ell}, \quad (10)$$

where $\rho_{\boldsymbol{s}_i,\boldsymbol{p}_\ell}$ (in dBm) is the power received from the sensor at position $\boldsymbol{s}_i$ by the node at position $\boldsymbol{p}_\ell$, that is the i-th entry of the vector $\boldsymbol{\rho}_\ell$, ρ_0 is the initial power (in dBm) set to 100, n_P is the path-loss exponent set to 4, $\|\boldsymbol{s}_i - \boldsymbol{p}_\ell\|$ is the Euclidian distance between the position $\boldsymbol{p}_\ell$ of the considered node and the sensor position $\boldsymbol{s}_i$, and $\varepsilon_{i,\ell}$ is the noise affecting the RSSI measures with $\sigma_{\boldsymbol{\rho}}$ its standard deviation. For the definition of $\boldsymbol{\psi}(\cdot)$, we consider the Gaussian kernel given by $\kappa(\boldsymbol{\rho}_u, \boldsymbol{\rho}_{u'}) = \exp\left(\frac{-\|\boldsymbol{\rho}_u - \boldsymbol{\rho}_{u'}\|^2}{2\sigma^2}\right)$, where σ is its bandwidth. The choice of the parameters η and σ is done using the 10-fold cross-validation technique [15], considering $\eta N_p = 2^v$ with $v \in \{-20, -19, \cdots, -1\}$ and $\sigma = 2^{v'}$ with $v' \in \{1, 2, \cdots, 10\}$. We consider cosine functions for the accelerations. Then, the trajectory shown on the left plot of Fig. 1 is generated by taking twice the primitive integral of the accelerations. The RSSI values collected by the moving target are calculated using (10). In order to simulate a noisy environment, both components of $\boldsymbol{\sigma}_{\boldsymbol{\gamma}}$ are taken equal to $0.01m/s^2$, and $\sigma_{\boldsymbol{\rho}}$ is taken equal to $1dB$. The left plot of Fig. 1 shows the environment setup, the real trajectory, and the estimated trajectory using the proposed method. Let the estimation error be evaluated by the root mean squared distance between the exact positions and the estimated ones, then the error is equal to $1.07m$.

We now compare our method to the one proposed in [10], where the authors perform tracking by combining the fingerprinting-based WKNN algorithm with a Kalman filter. This method consists of finding a position estimate using

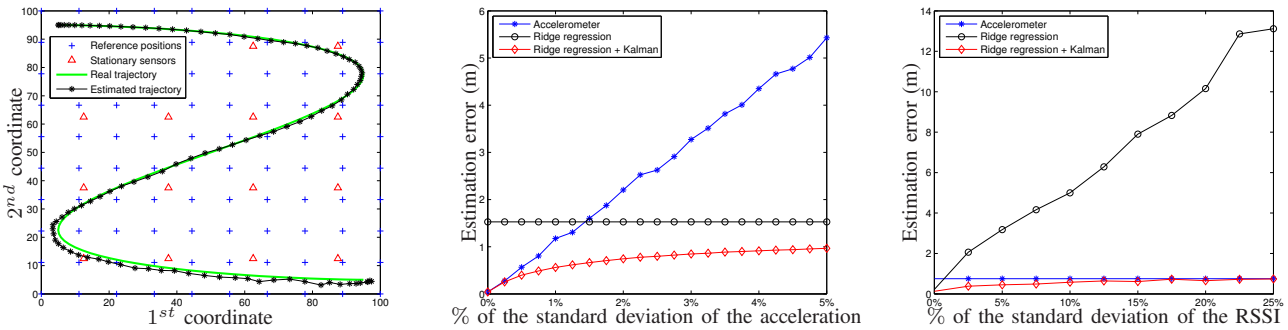

Fig. 1: Estimation of the trajectory using the proposed method on the left plot, estimation error as a function of the noise on the accelerations on the middle plot and as a function of the noise on the RSSI on the right plot.

WKNN, then enhances it by applying the Kalman filter. The number of neighbors K is taken equal to 8, and a second-order state-space model is used as in [10]. The algorithm is tested for the same setup as in the left plot of Fig. 1, and using the same values for $\boldsymbol{\sigma_\gamma}$ and $\sigma_{\boldsymbol{\rho}}$. The estimation error obtained is equal to $2.88m$. One can see that this error is significantly higher than the one obtained with our method.

We now propose to study the impact of the noises standard deviations $\boldsymbol{\sigma_\gamma}$ and $\sigma_{\boldsymbol{\rho}}$ on the estimation error. All results are averaged over 50 Monte-Carlo simulations. First, different percentages of the standard deviation of the acceleration are taken, going from 1% to 10%, along with a fixed $\sigma_{\boldsymbol{\rho}}$ equal to 5% of standard deviation of the RSSI measures. The standard deviation of the RSSI measured at the reference positions is equal to $10.79dBm$, leading to $\sigma_{\boldsymbol{\rho}} = 0.54dBm$. The results in the middle plot of Fig. 1 show that the ridge regression combined with the Kalman filter yields the best results, even when the noise on the accelerations is high. We then fix the value of $\boldsymbol{\sigma_\gamma}$ to 1% of the standard deviation of the acceleration, and consider several percentages of the standard deviation of the RSSI measures, going from 0% to 50%; in other words, $\sigma_{\boldsymbol{\rho}}$ varies from 0 to $5.40dBm$. The right plot of Fig. 1 shows that the ridge regression is highly affected by the noise variations, since RSSI measurements are used for the estimation, while tracking using only accelerometer information is not. However, the ridge regression combined with the Kalman filter yields the best results here as well, proving the effectiveness of using the Kalman filter.

VI. CONCLUSION

In this paper, we proposed a new method for target tracking in WSNs that combines ridge regression and Kalman filtering. Simulation results showed that our method allows accurate tracking, and is proved to be robust in the case of noisy data. It also outperforms tracking using a recently developed method based on the WKNN method and a Kalman filter. Future work will handle further improvements of our method, such as introducing a decentralized version.

VII. REFERENCES

[1] C. Gomez and J. Paradells, "Wireless home automation networks: A survey of architectures and technologies," *Communications Magazine, IEEE*, vol. 48, no. 6, pp. 92–101, 2010.

[2] P. Honeine, F. Mourad, M. Kallas, H. Snoussi, H. Amoud, and C. Francis, "Wireless sensor networks in biomedical: body area networks," in *Proc. 7th International Workshop on Systems, Signal Processing and their Applications*, Algeria, 09–11 May 2011.

[3] D. Li, K. Wong, Y. H. Hu, and A. Sayeed, "Detection, classification, and tracking of targets," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 17–29, 2002.

[4] E.-E.-L. Lau and W.-Y. Chung, "Enhanced rssi-based real-time user location tracking system for indoor and outdoor environments," in *International Conference on Convergence Information Technology*, 2007, pp. 1213–1218.

[5] S. Dias and M. Bruno, "Cooperative target tracking using decentralized particle filtering and rss sensors," *IEEE Transactions on Signal Processing*, vol. 61, no. 14, pp. 3632–3646, 2013.

[6] F. Mourad, H. Snoussi, M. Kieffer, and C. Richard, "Robust bounded-error tracking in wireless sensor networks," *16th IFAC Symposium on System Identification, SYSID, 2012*.

[7] T.-N. Lin and P.-C. Lin, "Performance comparison of indoor positioning techniques based on location fingerprinting in wireless networks," in *Wireless Networks, Communications and Mobile Computing, International Conference on*, vol. 2, 2005, pp. 1569 – 1574.

[8] S. Mahfouz, F. Mourad-Chehade, P. Honeine, H. Snoussi, and J. Farah, "Kernel-based localization using fingerprinting in wireless sensor networks," in *IEEE 14th Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2013, pp. 744–748.

[9] ——, "Decentralized localization using fingerprinting and kernel methods in wireless sensor networks." in *Proc. 21th European Conference on Signal Processing (EUSIPCO)*, 2013.

[10] E. Chan, G. Baciu, and S. C. Mak, "Using wi-fi signal strength to localize in wireless sensor networks," in *WRI Int. Conf. on Communications and Mobile Computing*, vol. 1, 2009, pp. 538–542.

[11] C. Saunders, A. Gammerman, and V. Vovk, "Ridge regression learning algorithm in dual variables," in *In Proceedings of the 15th International Conference on Machine Learning*. Morgan Kaufmann, 1998, pp. 515–521.

[12] G. Welch and G. Bishop", "An Introduction to the Kalman Filter," http://www.cs.unc.edu, UNC-Chapel Hill, TR95-041, February 2001.

[13] T. Hofmann, B. Schlkopf, and A. J. Smola, "Kernel methods in machine learning," *Annals of Statistics*, vol. 36, no. 3, pp. 1171–1220, 2008.

[14] A. Medeisis and A. Kajackas, "On the use of the universal Okumura-Hata propagation prediction model in rural areas," *Vehicular Conference Proceedings*, vol. 3, 2000.

[15] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *J. Royal Stat. Soc.*, vol. 36, no. 2, pp. 111 – 147, 1974.