WiFi Multifloor Indoor DCM Positioning

Rafael Saraiva Campos, Lisandro Lovisolo and Marcello L. R. de Campos

***Abstract*— Database correlation methods (DCM) are used to locate mobile stations (MS's) in wireless networks. A target radio-frequency (RF) fingerprint - measured by the MS to be localized - is compared with georeferenced RF fingerprints, previously stored in a correlation database (CDB). This paper focuses on the DCM positioning in multifloor indoor environments. In this scenario, the authors apply two combined techniques to reduce the search space inside the CDB, while improving the floor identification accuracy: i) unsupervised clustering using a single Kohonen Layer and ii) floor classification using committees of backpropagation artificial neural networks (ANN's), one committee per each floor. The effects of the proposed solution on the DCM positioning accuracy are experimentally evaluated using 46200 target fingerprints and a CDB with 924 reference fingerprints, containing Received Signal Strength (RSS) values of 136 WiFi 802.11b/g networks in a 12-floor building. The correct floor is identified in 91% of the samples, and is within 2 floors in 99% of the samples. The average positioning error is 4.7 meters and is below 5.5 meters in 75% of the samples.**

***Keywords*— Mobile Stations, WiFi Networks, Indoor Positioning, Radio-frequency Fingerprint, Kohonen Layer, Backpropagation.**

I. Introduction

There is a growing number of MS's equipped with built-in Global Positioning System (GPS) receivers. In open areas, GPS yields the highest location precision, but is usually unavailable in indoor environments. In this scenario, RSS based location techniques are used both in cellular and WiFi networks. However, in such environments, positioning using WiFi RSS values yields higher precision, due to the usually higher density of WiFi access points (AP's) in a indoor environment, in relation to cellular micro or even picocells.

RSS based DCM is a viable alternative for indoor WiFi positioning [1]. There is a wide variety of DCM solutions in the literature, but all share the same basic elements [2]. One of these elements is the CDB search space reduction technique, which has an impact both on the method's computational complexity and on the positioning precision. In this paper, two techniques are combined to reduce the search space within the CDB, while improving the correct floor identification accuracy: unsupervised clustering using a single Kohonen layer and floor classification using committees of backpropagation ANN's.

The remainder of this work is organized as follows: Section II introduces the basic elements of DCM; Section III provides a diagram of the proposed solution; Section IV describes the unsupervised clustering technique; Section V describes the floor classification procedure; Section VI details the experimental evaluation; and Section VII brings a brief conclusion.

Rafael Saraiva Campos (CEFET/RJ, PEE/COPPE/UFRJ), Lisandro Lovisolo (DETEL/FEN/UERJ) and Marcello L. R. de Campos (PEE/COPPE/UFRJ). Emails: rafaelsaraivacampos@gmail.com, lisandro@uerj.br, mcampos@ieee.org

II. Database Correlation Methods

DCM, also known as RF fingerprinting positioning, is a class of MS positioning methods that can be applied in any wireless network. Even though there is a wide variety of such methods, all present the same basic elements: RF fingerprints, a CDB, techniques to reduce the search space within the CDB, and the correlation of RF fingerprints.

A. RF Fingerprint

A RF fingerprint is a set of RF signal parameters. Those parameters are measured by the MS or by its anchor cells. Just like a human fingerprint, which carries the unique identification of a person, a RF fingerprint is expected to uniquely identify a geographic position.

A RF fingerprint can be classified as either a *target* (**TFing**) or *reference* (**RFing**) fingerprint. A **TFing** is the RF fingerprint associated with the MS which is to be localized, i.e., it contains signal parameters measured by the MS or by its anchor cells. The **RFing**'s are the RF fingerprints collected or generated during the training phase and stored in the CDB. Each **RFing** is associated with a 3-uple of geographic coordinates (x, y, z). The fingerprint structure used in this work, common to both **RFing**'s and **TFing**'s, is defined by the vector $\vec{F} = [\mathrm{RSS}_1 \ldots \mathrm{RSS}_N]$, where N is a constant which informs the number of WiFi 802.11b/g networks used in the position fix.

B. Correlation Database

The CDB is the set of **RFing**'s. The CDB is built during the DCM training phase [3], using radio propagation modeling, field measurements or a combination of both [4]. Each CDB entry is described by $\left(\vec{F}, x, y, z\right)$, where $\vec{F}$ is the **RFing** associated to the point defined by coordinates (x, y, z).

C. Techniques to Reduce the Search Space within the CDB

The search space or correlation space is a subset of the **RFing**'s stored in the CDB. The **RFing**'s in this subset are compared to the **TFing** to locate the MS. The geographic coordinates associated with the **RFing**'s in the search space are candidate solutions for the MS positioning problem. The CDB might be quite large and analyzing all RF fingerprints stored in it might be very time consuming. Therefore, all fingerprinting location techniques apply some method to reduce the search space within the CDB. As a consequence, the time required to produce a position fix is also reduced. Some of the techniques applied in the literature are deterministic filtering [5] and optimized search using genetic algorithms [6], both applied upon RSS maps built with empirical propagation models [7]. In [8], the search space is reduced by clustering the measurement points, i.e, the candidate solutions. This clustering is based on the identity of the q WiFi networks with the highest RSS at each measurement point.

D. Correlation of RF Fingerprints

The MS is assumed to be located at the point whose **RFing** has the highest correlation or similarity with the **TFing**. Comparison of the **TFing** and **RFing**'s can be carried out by calculating the distance between these fingerprints in the N-dimensional RSS space. If $\boldsymbol{C}$ is the set of **RFing**'s in the CDB and $\boldsymbol{D}$ is the set of **RFing**'s in the search space, then $\boldsymbol{D} \subseteq \boldsymbol{C}$. Let $\vec{F}$ be the **TFing** measured by the MS to be localized. The **RFing** $\vec{F}_k$ most similar to $\vec{F}$ is given by:

$$k = \underset{i}{argmax}\left[-\left(\vec{F}-\vec{F}_i\right)\left(\vec{F}-\vec{F}_i\right)^{\mathrm{T}}\right],\ \forall \vec{F}_i \in \boldsymbol{D} \qquad (1)$$

The MS estimated position is given by the coordinates associated with $\vec{F}_k$, obtained from the 4-uple $\left(\vec{F}_k, x, y, z\right)$ stored in the CDB.

III. Diagram of the Proposed Solution

Fig. 1 shows a diagram of the proposed solution, in the post-training phase or *on-line* phase. Initially, the **TFing** is transported to the principal components subspace, through principal components analysis (PCA) (step 1). Then, it is presented to both the Kohonen Layer and the committees of backpropagation ANN's (step 2). The Kohonen Layer identifies the cluster which the **TFing** belongs to. The committees of backpropagation ANN's identify the floor where the **TFing** was collected. The reduced search space will be given by the **RFing**'s present both in the current floor and the current cluster (step 3). Then, in the N-dimensional RSS space, the **TFing** is compared to all the **RFing**'s within the reduced search space (step 4). Finally, the process returns the coordinates (x, y, z) of the measurement point containing the **RFing** with the highest similarity with the **TFing** (step 5).

IV. Unsupervised Clustering using a Single Kohonen Layer

Due to the inherent complexity of the RF channel, it is not possible to know beforehand how the **RFing**'s - and consequently the measurement points where they were collected - will cluster. Under certain propagation conditions, measurement points far away from each other in the Euclidean three-dimensional space might have RF fingerprints which are close together in the N-dimensional RSS space (or the M-dimensional principal components subspace, where $M < N$). Therefore, for instance, it is not possible to ascertain that measurement points in the same floor will belong to the same cluster. So, there are no predefined targets during the training phase, and it is up to the classifier to identify, without

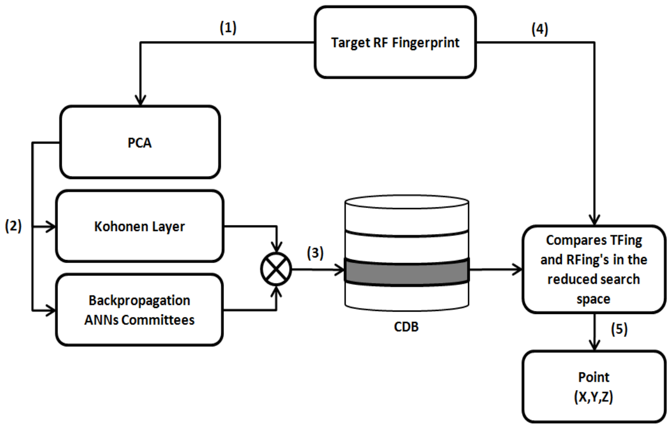

Fig. 1. Diagram of the Proposed Solution.

supervision, how the **RFing**'s will cluster together. Note that, as each **RFing**'s is georeferenced, i.e., is associated with a measurement point with known coordinates, by grouping the **RFing**'s in the N-dimensional RSS space, the classifier is indirectly grouping the measurement points in the Euclidean three-dimensional space [9].

The Kohonen layer used in this work is one-dimensional with a neighbor radius equal to zero, which means that only the winner neuron is activated. This classifier has an input layer and a competitive layer, as shown in Fig. 2. For each input vector $\vec{X} = [X_1\ X_2 \ldots X_N]$, the winner neuron at the competitive layer is that whose synapse $\vec{W}_i$ is the most similar to the input vector $\vec{X}$. The output of the winner neuron is activated (y_i=1), while the outputs of all other neurons remain equal to zero (y_j=0, $\forall j \neq i$) [10]. The similarity measure used was:

$$u_i = -\left[w_{i,0} + \left(\vec{X} - \vec{W}_i\right)\left(\vec{X} - \vec{W}_i\right)^T\right], \forall i \in [1, 2, \ldots, N_c] \quad (2)$$

where $w_{i,0}$ is the i-th neuron conscience bias, which is defined by Eq. (4), and N_c is the number of neurons in the Kohonen layer.

The initialization of the synaptic weights is critical for the algorithm convergence. The technique used for the initialization is described in Section VI. In order to distributed the training approximately evenly among all neurons in the competitive layer, conscience is used [11]. With conscience, neurons which are constantly winning receive a progressively decreasing negative *bias*. This allows less trained neurons to become more similar to the input vectors. With the conscience mechanism, neurons in the Kohonen layer naturally represent approximately equal amount of information [12]. Individual decreasing learning steps per neuron are also used [10].

V. Floor Classification using Committees of Backpropagation ANN's

The Kohonen Layer is used to cluster the target RF fingerprints in a unsupervised manner. After training, each neuron in the Kohonen layer maps one cluster within the CDB. However, a cluster might span several floors, as shown in Fig. 3, and, in multifloor indoor positioning, it is very important to correctly

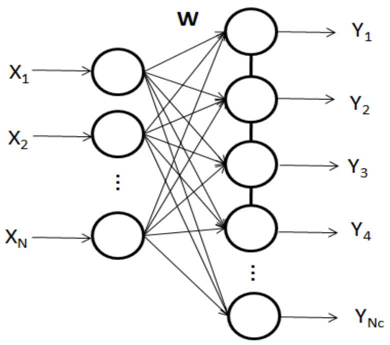

Fig. 2. ANN with one competitive layer.

identify the floor were the MS is located, before estimating the MS 2D position within that floor. So, after selecting the cluster using the Kohonen Layer, the proposed solution uses committees of backpropagation ANN's to identify the floor where the MS might be located, within the selected cluster.

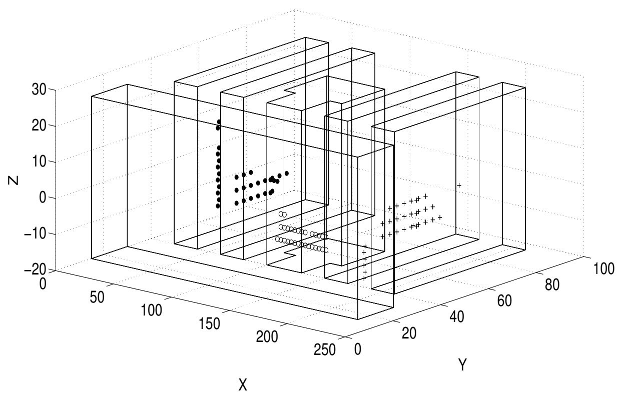

Fig. 3. Examples of clusters defined by the Kohonen Layer in [9]

Each one of the F floors has a committee with N_a backpropagation ANN's. Each ANN was trained as a binary classifier, identifying if a given RF fingerprint belongs to the current floor or not. Each classifier is identified by $g_{n,f}$, where $n \in [1, \ldots, N_a]$ and $f \in [1, \ldots, F]$. At each floor, the outputs of the single classifiers are combined into a unique joint voting classifier g_f^v, whose output is the class the majority of the single classifiers voted for. The output of the joint voting classifier is expected to have a lower variance than the outputs of the single binary classifiers [13].

VI. Experimental Evaluation

A. Experiment Setup

The WiFi RSS measurement campaign was carried out in the 12 floors of Principal Joao Lyra Filho Pavilion at the University of Rio de Janeiro State (UERJ). The software used to collect the WiFi scans was *NetStumbler* version 0.4, which was run in a Toshiba A75-S211 laptop with a Atheros AR5005GS built-in 802.11b/g adapter. *NetStumbler* forces the WiFi adapter to carry out a passive scan of 802.11 networks, i.e., without sending probe requests. During the passive scan, the WiFi adapter remains a certain time period on each channel, waiting to receive a beacon. The beacon, which is sent by every WiFi access point (AP), contains the network identifier (SSID - *Service Set ID*) and the AP MAC (*Medium Access Control*) address. For each detected AP, *NetStumbler* stores the MAC, SSID, carrier number, noise level and signal-to-noise ratio. The laptop was placed over a wheeled table, and at each of the 924 measurement points the WiFi adapter collected between 180 and 240 WiFi scans, at a rate of one per second. Each WiFi scan contains data from several AP's. Fig. 4 shows a perspective spatial view of the measurement points positions in the UERJ main building.

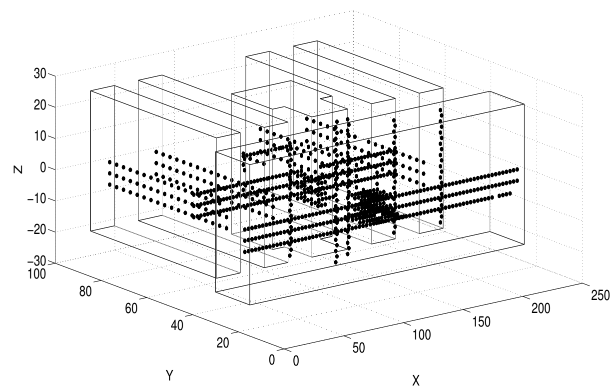

Fig. 4. Measurement points at UERJ building.

Each measurement point is identified by a unique **RFing** containing the mean RSS values of each detected WiFi network. At each measurement point, the mean RSS values per WiFi network are calculated using all WiFi scans carried out in that particular point. If a WiFi network is detected in less than 10% of the WiFi scans at a given point, its mean RSS value is assumed to be zero. A total of $N = 136$ WiFi networks were detected, resulting in a training set matrix of size 924×136.

The first 50 WiFi scans at each measurement point were used as input vectors of the test set. Considering all measurement points, a total of 924×50 WiFi scans was selected for the test set, resulting in a test set matrix of size 46200×136.

Prior to presenting the **RFing**'s to the classifier, the RSS values were converted to the logarithmic scale in order to compress them to the numerical range -120 to -30 dBm. Then, as shown in Fig. 1, PCA was applied to reduce the input vectors dimension. PCA generates a new set of mutually orthogonal variables called principal components (PC's). Firstly, the training set, with M input vectors, is translated by extracting the sample mean at each dimension, obtaining matrix $\mathbf{B} = \left[v_{i,j} - \bar{V}_j\right]_{i=1,\ldots,M;j=1,\ldots,N}$, where $v_{i,j}$ is the j-th WiFi network RSS value at the i-th scan, and $\bar{V}_j$ is j-th network RSS sample mean. Let $\vec{U}_j$ be the j-th eigenvector of the covariance matrix of $\mathbf{B}$. The PC's matrix of the training set is given by $\mathbf{P}_{M\times N} = \mathbf{B} \cdot \left[\vec{U}_j\right]_{j=1,\ldots,N}$.

The columns in $\mathbf{P}$ are sorted in decreasing variance order, each one corresponding to a PC. To reduce the training patterns dimension, only the first p PC's are held. In this paper, p was selected so that at least 99% of the training set total variance was preserved, resulting in $p = 78$.

The sample means and the matrix $\left[\vec{U}_j\right]_{j=1,\ldots,N}$, obtained during application of PCA to the training set, are also used to project each test set vector into the PC's subspace. Just like in the training set, only the first p PC's are held.

B. Training the Single Kohonen Layer

The Kohonen Layer synaptic weights are defined in the PC's subspace. So, there is a $N_c \times p$ matrix of synaptic weights, where N_c is the number of clusters and p is the number of PC's that are kept ($p = 78$). The theoretical optimum value of N_c, as a function of N_p, is calculated in [9]. For $N_p = 924$, one has $N_c = 30$.

For the initialization of the synaptic weights in the PC's subspace, the maximum variance dimension (the first PC) has been divided into N_c equal length sections. In each section, a input vector whose first PC value is equal to the section median is selected. The selected input vectors are the initial values for the N_c neurons synaptic weights [14].

During the Kohonen layer training, individual decreasing learning steps per synapse were used, as defined by:

$$\alpha_i(n) = \alpha_0 \exp(-n/N_0) \tag{3}$$

where $\alpha_i(n)$ is the i-th synapse learning step at instant n, $\alpha_0 = 0.45$ e $N_0 = 120$.

Conscience was used in the training phase. Conscience is implemented by adding a negative bias to the neuron similarity function, which was defined by Eq. (2. The negative bias is given by:

$$w_{i,0} = \phi^2 \{0.5 [1 - \tanh(k(p_i - p^*))] - 1\} \tag{4}$$

where ϕ is equal to the diameter of the single class (i.e., assuming that all vectors belong to only one class and finding the highest distance between any pair of input vectors) in the PC's subspace, $k = 4.5$, p_i is the percentage of times the i-th neuron is trained (number of times the neuron was trained or won over the number of input vectors presented to the classifier) and $p^* = 1/N_c$.

At each training epoch, all 924 training patterns are presented to the Kohonen layer. The training continues until a maximum number of epochs has been reached (20 epochs), or when the maximum variation of the synaptic weights between two consecutive epochs is below a certain threshold (0.0001).

C. Training the Committees of Backpropagation ANN's

Each floor has a joint majority voting classifier composed by a committee of $N_a = 13$ ANN's. So, there are 12 committees (one per floor) and 156 single binary classifiers. All ANN's have the same topology, with $p = 78$ inputs, $q = 10$ neurons in the hidden layer and 1 neuron in the output layer. The activation function of all neurons is the hyperbolic tangent. The training method is *Levenberg-Marquardt* [15], with mean square error (MSE) as the performance function. The optimum size of the hidden layer was determined by increasing the number of neurons during the training phase, until no further relevant reduction in the MSE was detected.

Training was carried out in MATLAB. The training continues until one of the following conditions is met: i) a maximum number of epochs is reached (50 epochs); ii) a goal is met for the MSE (equal to or below 0.01); iii) a maximum of number consecutive validation fails occur (6 fails).

Of the total $N_t = 924$ **TFing**'s in the CDB, N_f belong to the f-th floor, and $(N_t - N_f)$ belong to other floors. As there are 12 floors, $N_f < (N_t - N_f)$. So, to prevent biasing the training of the classifiers in the committee g_f^v, the **TFing**'s belonging to the f-th floor are repeated $\lfloor \log_2 [(N_t - N_f)/N_f] \rfloor$ times. So, the number of **TFing**'s in g_f^v that belong to the f-th floor will be approximately the same of the remainder **TFing**'s in the CDB. Therefore, each committee has a different training set.

Validation vectors are used to prevent *overtraining* [10]. They stop training early if the network performance on the validation vectors fails to improve or remains the same for a maximum number of consecutive epochs. Within the training set of each ANN, 10% of the **TFing**'s are randomly selected for the validation set. Therefore, not only each committee has a different training set, but also each ANN within each committee.

The output of each ANN is within the range $[-1, +1]$. Each committee will consider that a classifier has voted 1 (**TFing**'s belong to the current floor) if its output is positive, and 0 if otherwise. If two or more committees have voted 1 (a **TFing** cannot belong to more than one floor) , the committee with the highest cumulative value for the positive outputs of its classifiers wins.

D. Experiment Results

Fig. 5 shows the experimental Cumulative Distribution Function (CDF) of the positioning error of the following methods: (I) pure DCM; (II) DCM with ANN's committees; (III) DCM with Kohonen Layer; (IV) DCM with Kohonen Layer and ANN's committees. It can be seen that, methods (I) and (II) have approximately the same error distribution, as well as methods (III) and (IV).

Table I shows that, while the use of ANN's committees (method II) has increased the floor identification accuracy from 78% to 91% in relation to pure DCM (method I), its has not significantly improved the positioning error. However, the use of Kohonen Layer (method III) has reduced the average error by almost 40% in relation to pure DCM (method I). By combining the two techniques (method IV), both benefits are present: increased floor identification accuracy (up 13% in relation to pure DCM) and higher positioning accuracy (average error down 40.5% in relation to pure DCM).

The floor estimated by the proposed method (method IV) is within two floors of the correct floor in 99% of the cases, against 97% of the cases in pure DCM (method I).

E. Comparison with Published Results

There is a wide list of papers about RF indoor positioning problem, but, surprisingly, the authors have found only a few

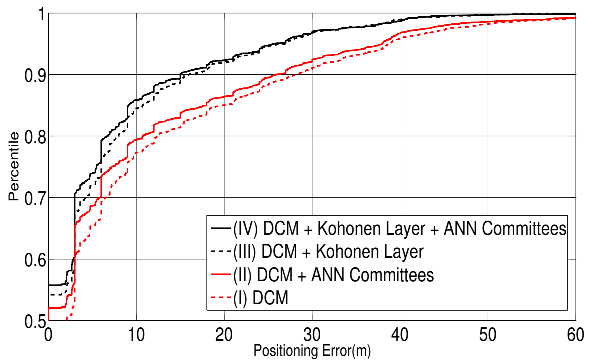

Fig. 5. Positioning Error CDF.

TABLE I

DCM POSITIONING ERROR IN METERS AND FLOOR IDENTIFICATION ACCURACY.

Method	Avg Error	75th Percentile	90th Percentile	Floor Id. Accuracy
I	7.9	9.0	28.0	78%
II	7.1	7.2	26.0	91%
III	5.0	6.0	16.0	84%
IV	4.7	5.5	15.0	91%

addressing multifloor scenarios [16] [17] [18] [19].

In [16], a Global System for Mobile Communications (GSM) fingerprinting indoor localization system for multifloor positioning was proposed, achieving a 73% floor identification accuracy. The estimated floor was within two floors of the correct floor in 97% of the cases. The system has been tested in three tall buildings (9, 12 and 16 floors) and the number of fingerprints collected per floor ranged from 30 to 130.

In [17], the authors collected RSS measurements in 30 points, 21 at the first floor and 9 at the second floor. In [18], a single-phase location determination system (no training phase) was proposed and tested, also in a two-store building. As the buildings had only two floors, the multifloor effect in the positioning error was diminished.

In [19], the authors proposed a solution that combined trilateration and scene analysis method, reporting a 100% floor identification accuracy. However, in the 8 floor building where the method was tested, only 57% RF fingerprints were collected.

In [20], the positioning error for the 75th percentile was 4.7 meters when the **TFing**'s were built using mean RSS values of 20 samples per measurement point. When just one sample was used per **TFing** (as in this work), the positioning error for this same percentile rose to 6.1 meters. This precision is approximately the same achieved by Method IV, as shown in Table I.

VII. CONCLUSION

In this work, the authors proposed the use of two combined techniques to improve DCM positioning error in WiFi networks in multifloor indoor environments: a single Kohonen Layer, to cluster the RF fingerprints in the CDB and reduce the search space, and committees of backpropagation ANN's, to further reduce the search space and improve floor identification accuracy. The proposed combined techniques achieved 91% floor identification accuracy and 4.7 meters average positioning error.

REFERÊNCIAS

[1] K. Thomas et al, "On-demand fingerprint selection for 802.11-based positioning systems," in *Proc. of 2008 IEEE international Symposium on a World of Wireless, Mobile and Multimedia Networks*, Newport Beach, USA, June 2008, pp. 1–8.

[2] H. Laitinen, J. Lahteenmaki, and T. Nordstrom, "Database Correlation Method for GSM Location," in *Proceedings of IEEE 53rd Vehicular Technology Conference*, Rhodes, Greece, May 2001, pp. 2504–2508.

[3] Sinan Gezici, "A Survey on Wireless Position Estimation," *Wireless Personal Communications: An International Journal*, vol. 44, no. 3, pp. 263–282, February 2008.

[4] J. Zhu and G. D. Durgin, "Indoor/Outdoor Location of Cellular Handsets Based on Received Signal Strength," *Eletronic Letters*, vol. 41, no. 1, pp. 24–26, January 2005.

[5] R. S. Campos and L. Lovisolo, "A Fast Database Correlation Algorithm for Localization of Wireless Network Mobile Nodes using Coverage Prediction and Round Trip Delay," in *Proceedings of IEEE 69th Vehicular Technology Conference*, Barcelona, Spain, April 2009.

[6] M. J. Magro and C. J. Debono, "A Genetic Algorithm Approach to User Location Estimation in UMTS Networks," in *EUROCON - The International Conference on Computer as a Tool*, Warsaw, Poland, September 2007, pp. 1136–1139.

[7] M. Hata, "Empirical Formula for Propagation Loss in Land Mobile Radio Services," *IEEE Transactions on Vehicular Technology*, vol. 29, no. 3, pp. 317–325, August 1980.

[8] M. Youssef et al, "WLAN Location Determination via Clustering and Probability Distributions," in *Proc. of the First IEEE International Conference on Pervasive Computing and Communications*, Seattle, USA, March 2003, pp. 143–150.

[9] R. S. Campos, L. Lovisolo, and M. L. R. de Campos, "Search Space Reduction in DCM Positioning using Unsupervised Clustering," in *Proceedings of IEEE 10th Workshop on Positioning, Navigation and Communication (WPNC'13)*, Dresden, Germany, March 2013.

[10] Simon Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall PTR, 1994.

[11] D. DeSieno, "Adding a conscience to competitive learning," in *Proc. of 1988 IEEE International Conference on Neural Networks*, Sand Diego, USA, July 1988, pp. 117–124.

[12] R. G. Silva, "Improved Self-Organizing Map Performance Applied To Process Monitoring," in *Proc. of 6 Congreso Interamericano de Computacion Aplicada a La Industria de Procesos*, Puebla, Mexico, October 2003.

[13] Friedrich Leisch and Kurt Hornik, "Arc-lh: A new adaptive resampling algorithm for improving ann classifiers," in *Advances in Neural Information Processing Systems*. 1997, MIT Press.

[14] M. Belal and A. Daoud, "A New Algorithm for Cluster Initialization," in *Proc. of the Second World Enformatika Conference*, Istanbul, Turkey, 2005.

[15] J. J. Moré, "The Levenberg-Marquardt Algorithm: Implementation and Theory," *Lecture Notes in Mathematics*, , no. 630, pp. 105–116, 1977.

[16] A. Varshavsky et al, "The SkyLoc Floor Localization System," in *Proceedings of the Fifth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom'07)*, White Plains, USA, March 2007, pp. 125–134.

[17] A. S. Al-Ahmadi et al, "Multi-Floor Indoor Positioning System using Bayesian Graphical Models," *Progress In Electromagnetics Research B*, vol. 25, pp. 241–259, 2010.

[18] A. S. Al-Ahmadi et al, "Single-Phase Wireless LAN Based Multi-Floor Indoor Location Determination System," in *Proceedings of IEEE 17th International Conference on Parallel and Distributed Systems (ICPADS)*, Tainan, Taiwan, December 2011, pp. 1057–1062.

[19] H. H. Liu and Y. Yang, "WiFi-Based Indoor Positioning for Multi-Floor Environment," in *Proceedings of 2011 IEEE Region 10 Conference (TENCON 2011)*, Bali, Indonesia, November 2011, pp. 597–601.

[20] P. Bahl and V. N. Padmanabhan, "RADAR: An In-Building RF-based User Location and Tracking System," in *Proceedings of 19th Annual Joint Conference of the IEEE Computer and Communications Societies*, Tel Aviv, Israel, March 2000, pp. 775–784.