Indoor Position Method Using Wi-Fi

Luis Fernandes, Fábio Barata, Paulo Chaves

INOV- Inesc Inovação
Rua Alves Redol ,9, 1000-029, Portugal

Abstract. The main objective of this paper is to present an indoor position method, developed for museums and hospitals, in order to known the position of persons and equipments. The method is composed by a particle filter algorithm to be embedded in smartphones or tags. The main goal is to develop a light method to run by not very powerful microprocessors, energized by batteries. The position method presented is based on a Wi-Fi structure, to lighten the use of the communication network, the position is generated by each device from raw data received locally.

Keywords. Wi-Fi, fingerprinting, algorithm, particle filter

1. Motivation and Context

The work presented in this paper is an output from two projects, one applied to hospitals, in which the position of persons and assets is used to control the proliferation of infectious diseases. The intention is to find contacts, between persons and infer the source of a disease.

The other project is dedicated to museums, the position of the persons is used by an application to show and describe, to the user, the assets which are nearby namely statues and paintings.

The use of the Wi-Fi signal level, to obtain the position is a requirement from these applications, it is required that the position should be given by the technologies offered by smartphones. There are several component sensors embedded in a smartphone, that could have been used to determinate the position, besides the Wi-Fi, namely the Bluetooth, NFC receiver, camera and to improve the positioning the magnet sensor and the gyroscope.

Published in "Proceedings of the 11th International Symposium on Location-Based Services", edited by Georg Gartner and Haosheng Huang, LBS 2014, 26–28 November 2014, Vienna, Austria.

After short study it was chosen the Wi-fi, Bluetooth and the camera, for the museum application and only the Wi-Fi for the hospital. The museum environment relies on the assistance of visitors to use the smartphones, but in the case of hospital the user interaction should be minimal and the smartphones should evolve to tags. This paper describes only the method that was used with the Wi-Fi technology.

2. Introduction

This work uses the Received Signal Strength Indicator (RSSI) to infer the position, the RSSI is the radio signal power present at the receiver at a distance (d) from the transmitter. In general the RSSI decreases proportionally with the distance [1][6], if the relationship between the signal level and the distance is known, it is possible to deduce the distance between two devices. The advantages of using the RSSI with radio technologies like Wi-Fi, is that it requires no hardware changes and the functionality of reading the RSSI is offered by almost of the communication devices. The localization can be produced by simply adding software to the hosts and does not require any kind of synchronization between receiver and emitter. The main disadvantaged is the unpredictability of RSSI levels for most indoor situations, due to the phenomena of multipath and attenuation, derived from walls or other objects that are between the emitter and the receiver. These phenomena can be overcome with other techniques, if we consider that a group of sources generate a single vector of RSSI for each position it is possible to determinate a position by comparing two vectors.

2.1. Measuring principles

There are several methods to process the information given by the sensors, the main methods are; geometric methods, fingerprinting and the proximity methods.

The most common geometric methods are multi-lateration [2][1] using ToA/ToF (Time of Arrival/Time of Flight), Round Trip Time (RTT), TDoA (Time Difference of Arrival)[5] or RSSI, with these methods the target location is estimated by measuring its distance from multiple reference points.

In the case of ToA/ToF the distances are given by the travel time between synchronized transmitter and receiver devices. Having into account the speed of the light, the receiver can find the time of arrival by subtracting the time at which the signal was transmitted from the time at which the signal was received.

For the RTT method [2] the time taken by the signal to travel from a transmitter to a receiver and back is measured. RTT avoids the need for time synchronization between the transmitter and the receiver, allowing its application in uncoordinated mesh networks with the advantage of low complexity and cost. The TDoA uses the time difference of arrival between multiple synchronized transmitters measured at the receiver.

The use of the RSSI in multi-lateration is based on the principle that the RSSI decreases with the distance, but in practice there are many other factors that make this method impracticable in most indoor situations. In (1) is described the model Log-distance Path Loss, in this model the received power (dBm) at a distance d (in meters) depends on the distance d, the path loss exponent α, and the power receive at 1 meter from the transmitter. X_σ represents a Gaussian random variable with zero mean and standard deviation of σ. The parameters α and σ define the statistical model and are heavily dependent on the environment [6].

$$Pr(d) = Pr_0 - 10\alpha log(d) + X_\sigma \qquad (1)$$

To implement an indoor system, with a minimum performance, these factors should be reflected in a huge central data base, requiring that the position is calculated in a central machine. This method also heavy for the local communication network that supports the positioning, the communication of the raw data from the targets to the central processing machine can became too heavy.

The proximity or Cell of Origin (CoO) [2][1] method is used to set the position of a target merely by its presence in a particular area, based on the RSSI. The procedure consists in simple forwarding the position of reference point where the strongest signal is received. The accuracy of CoO is dependent on the density of references and signal range. CoO is a simple positioning method used for applications with low accuracy requirements.

The fingerprinting is a method [1][2] that maps the measured data, example the RSSI, the magnetic field, audio signal or images, to a known grid-point covering an area of interest. Typically it consists of two phases, in the first phase, the RSSI received from fixed stations are measured at a number of grid-points and added to a database. In the operation phase the current measured RSSI are compared for the best agreement with a database.

2.2. Location Estimation Algorithms

This paragraph summarizes some of the methods commonly used to infer location from RSSI measurements.

K Nearest Neighbor Method (kNN) [2]. The nearest neighbor methods are deterministic algorithms they require only a set of constant location fingerprints which includes mean and standard deviation vectors of RSSI. The kNN method uses the RSSI to search for k closest matches of known locations in signal space from a previously-built database according to root mean square errors principle.

The probabilistic approach models a location fingerprint with conditional probability and utilizes the Bayesian inference concept to estimate the target locations. This approach presumes a priori knowledge of the probability distribution of the target's location.

An exmaple method considers positioning as a classification problem. Assuming that there are n location candidates L1, L2,...,Ln and Y is the observation vector, the following decision rule can be obtained[5]:

If p(Li |Y) > p(Lj |Y), for i, j=1,2, . . . , n , j≠ i. choose Li (2)

Here, p(Li|Y) is the posterior conditional probability, that the mobile is in location Li, given the observation vector Y. Also assumes that p(Li) is the probability that the target is in location Li .

Support Vector Machine Methods (SVM)[3][5]. To estimate the dependency between the RSSI fingerprint and the location from the observations, this approach does not require detailed properties of the dependency such as the propagation model as is in the probabilistic method. The strength of SVMs algorithm lies in its ability to generalize classification which minimizes the test error or the classification error for the data after the training period. The learning machine could be trained correctly by learning from a small training set and creating a sufficient structure for data classification without memorizing or over fitting the training samples.

This method particle filter is a powerful Bayesian algorithm. The particle filter robustness lies in the ability to handle non-linear system with non-Gaussian noise. The ability to incorporate a kinematic model of the moving target, in its probability model, makes this method a naturally suitable for sensor data fusion. The disadvantage is that it requires relatively high computational power.

2.3. The Location Fingerprint

A fingerprint based on RSSI is the basis for representing a unique position or location. It is created under the assumption that each position or location vector ($\mathcal{L}$), inside a confined area, has a unique signature $\mathcal{F}$[3]. The location and fingerprints are maintained in a database and used during the on-line phase to estimate the location. To create a data base of fingerprints, a number of sample vectors of RSSI are collected over a window of time for each position, 10 to 50 samples, from which is calculated de average value and the standard deviation for each AP. Extra fingerprint information such as standard deviation for each RSSI element, may be added into the data base fingerprint as another vector. For N access points, that can be heard at a location, a location fingerprint can be expressed as vector:

$$\mathcal{F} = (\rho_1, \dots, \rho_N) \qquad (3)$$

Where each ρ_i is an average RSSI for the position location $\mathcal{L}$ for the APi.

The location vector and the standard deviation vector can be expressed as:

$$\mathcal{L} = \{(x, y, z) |\ x, y \in \mathbb{R}^2, z \in \mathbb{N}\} \ (4)$$

$$\mathcal{D} = (\sigma_1, \dots, \sigma_N) \qquad (5)$$

Each σ_i is the standard deviation RSSI for the position location $\mathcal{L}$ for the APi, (x, y, z) is the coordinate of a position inside the interest area, in meters for x, y, and z represents the floor.

Another approach to represent fingerprint information is to estimate the probability distribution of the $\mathcal{F}$ signature. This approach is referred to as the probabilistic approach since it is assumed that the location fingerprint is described by a conditional probability. The added element to location fingerprints is the probability distribution estimated for the RSSI signature at a given $\mathcal{L}$. The location fingerprint becomes a conditional probability distribution of the form $P(Y|\ \mathcal{L})$ where Y de notes the observation vector of RSSI at location $\mathcal{L}$.

3. The particle filter

This chapter describes the particle filter for localization as a Bayesian approach. In order to clearly describe the problem, some terms should be clarified. Target is defined as an entity (e.g.: object, person) from which the state is being estimated. State, is defined as the collection characteristics of the target (location, velocity, direction) and the measurements are the observed phenomena obtained from a sensor which carries information about the state, in this case de RSSI measurements. The particle filter estimates the target state, based on the observed phenomena or measurements.

The evolution of the target state and measurements, during localization, can be seen as statistical Hidden Markov Model (HMM). The parameter Xt describes the state, of target, at time t, X is hidden and cannot be measured directly. The parameter Yt depicts the measurement at time t, Y can be observed directly. One can only estimate the state X from the observed measurement Y.

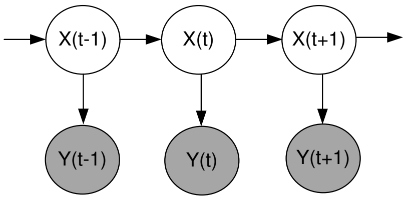

Figure 1. Representation of a Hidden Markov Model

The movement of the target state X_t and the measurement Y_t are defined by a discrete-time stochastic filtering model, composed by two equations the state equation and the measurement equation [7][1],

$$X_t = f_{t-1}\,(X_{t-1}, n_{t-1}) \qquad (6)$$

$$Y_t = h_t(X_t, e_t) \qquad (7)$$

The functions ft() and ht() are unknown, possible non-linear time varying functions, n_t and e_t are independent distributed noise.

A particle filter is an implementation of the formal recursive Bayesian filter using (sequential) Monte Carlo methods. It approximates the posterior probability to a finite number of discrete samples with associated weights, called particles. The particles are concrete instantiations of the state at time t, with the probability given by the weight ω_t^i. The posterior distribution of the state can be approximated by the (8) (when $N \uparrow \infty$), where X_t^i is the i-th particle, with 1<i<N, ω_t^i the weight of the particle i and N the number of particles.

$$p(X_t|Y_t) \approx \sum_{i=1}^{N} \omega_t^i \delta(X_t - X_t^i) \qquad (8)$$

The particle set is defined as χ_t.

$$\chi_t := X_t^1, \cdots, X_t^N \qquad (9)$$

The weight ω_t^i is called the measurement probability or the likelyhood observation probability, it is the probability of a state X_t^i that received the measurement Y_t. It is the probabilistic representation of $h_t(X_t, e_t)$.

The likelihood observation function $p(Y_t|X_t)$ or measurement model is an important part of the particle filter algorithm, this function should describe as accurate as possible the reality of the observation phenomena. In literature are some models that may suit an application, in this text we will reference just some simple models based on the distance between the signature vector $\mathcal{F}^i$ of the particle i and the measurement vector Y. The weight w_t^i should get larger as the measurement vector approximates to the signature vector [4].

$$w_t^i = \frac{1}{||\mathcal{F}^i - Y_t||} \qquad (10)$$

Some examples of norms $||z||(z \in \mathbb{R}^{n_z})$ are:

$$||z||_p = \left(\sum_{i=1}^{n_x} |z_i|^p\right)^{\frac{1}{p}} \text{ p-norm} \qquad (11)$$

$$||z||_{mp} = \left(\sum_{i=1}^{n_z} \frac{1}{w_i} |z_i|^p\right)^{\frac{1}{p}} \text{ modified p-norm} \qquad (12)$$

$$||z||_\infty = \max_i(|z_i|) \text{ infinity-norm} \qquad (13)$$

Finally w_t^i is normalized making

$$\omega_t^i = \frac{w_t^i}{\sum_{j=1}^{N} w_t^j} \qquad (14)$$

To estimate the target state location T_t, at t time, there are some methods based on the normalized values of ω_t^i and using the position location $\mathcal{L}_t^i$ vector, (14) and (15) are the two most used methods, other mixed methods may be used.

$$\overline{T}_t = \sum_{i=1}^{N} \mathcal{L}_t^i . \omega_t^i \quad \text{mean value} \ (15)$$

$$T_t = \mathcal{L}_t^{\max_i(\omega_t^i)} \quad \text{location of the max } \omega_t^i \text{ value} \qquad (16)$$

4. Application of the Particle Filter

4.1. The use of a probabilistic method

In order to improve the localization algorithm, it was chosen the probabilistic method, where the target and the particles of the filter are constrained by the following conditions. The particles can only travel inside of a map or network composed by nodes (where the particles are located) and connections that permit the particles to travel between nodes. This approach presumes a priori knowledge of the probability distribution of the target's location.

The figure 2 illustrates a network of nodes (in green), with connections between nodes (in red). The nodes are placed in positions where there is more probability to find a target and the connections are not placed randomly but enter in account the possible movements of the target. The particles are free to travel among any adjacent nodes, for each observation measurement, it means that $p(X_t|Y_{t-1})$ may have a probability of zero for most of the situations.

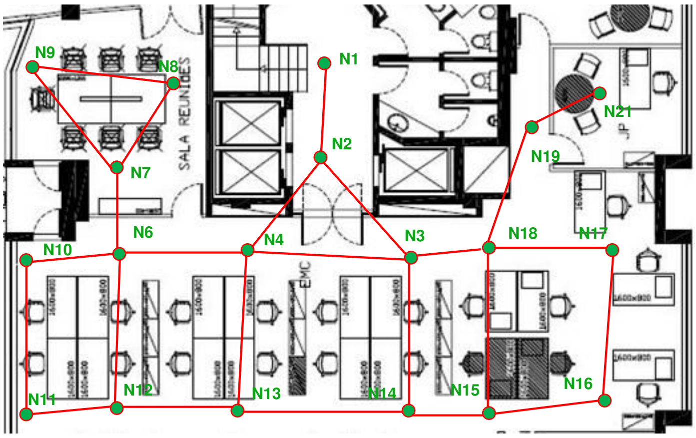

Figure 2. A network of nodes and connections

The network of nodes with fingerprinting introduces some extra variables to the process but eases the implementation of the algorithms and the complexity of the database of the fingerprinting. The network is composed by a fixed number of nodes, Nnodes, each node may have a number of cles, np_t^n at a given time t and to each node is attributed a weight W_t^n . The weight, W_t^n is dependent on the number of particles inside of the node and their weights. The nodes also have other attributes like the position in space and the signatures.

The fingerprint set is defined as Γ and is composed by N_{nodes} nodes.

$$\Gamma := \upsilon^1, \cdots, \upsilon^{N_{nodes}} \qquad (17)$$

Each node is composed by the following quintuple;

$$\upsilon^n = (\mathcal{L}^n, \mathcal{F}^n, \mathcal{A}^n, np^n, W^n) \qquad (18)$$

$$\mathcal{L}^n = \{(x, y, z) |\ x, y \in \mathbb{R}^2, z \in \mathbb{N}\} \qquad (19)$$

$$\mathcal{F}^n = (\rho_1, \dots, \rho_{NumberOfAP}) \qquad (20)$$

$$\mathcal{A}^n = \{(\wp^1 \dots \wp^{nbc}) | \wp^i \in \Gamma\} \qquad (21)$$

Where (18) is the quintuple representing the node υ^n, (19) is the location vector of the node υ^n, (20) is the fingerprint or signature vector and ρ_i represents the average value of RSSI for the APi, (21) is a group of pointers $\wp^i$, which point to the nodes connected to the node υ^n.

4.2. The motion model

The motion model is a representation of the target's kinematics behavior, it is used to construct the transition probability $p(X_t|Y_{t-1})$ which has an important role for the prediction step in the particle filter.

To introduce a motion model, the distance between the nodes, which is dependent on the observations sample frequency and the speed of the target, should be chosen. From practical tests it was found that a person can travel (indoors) at maximum 6m/s with a mean value of 3m/s, which requires that a system with a sample frequency, of the observed measurements, of 1Hz should have the nodes distanced of about 3m.

Another characteristic of this model is that the target can change direction instantly (in about 1s, corresponding to the sample rate) meaning that the motion of the particles have the same direction probability in all directions.

The figure 3 shows all possible movements of a target/particle in N1, it may stay at the same place or move to node N2 or to node N4, all other movements are forbidden for an observable measurement. The direction depends on the direction of the adjacent nodes and the speed on the distance between nodes and sample frequency. For a particle in N1 may have de speed of $v = 0$, $v = D1/T$ or $v = D2/T$ (where T is the period of sample)

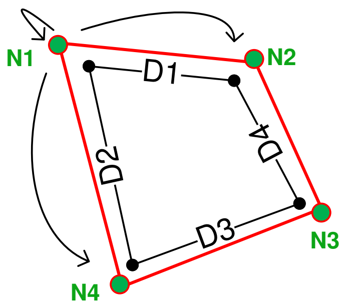

Figure 3. Movements of a particle at N1

4.3. Particle filter implementation

The algorithm (22) describes the implementation of a particle filter with the modifications and adaptations necessary to use with a network of nodes.

$(\Gamma_t, i_{maxW}) = \text{particlefilter}(\Gamma_{t-1}, Y_t)$ (22)

1: $\Gamma_t^* = \Gamma_{t-1}$ //use an auxiliary network of nodes

2: //assign zero to the number of particles and weight of the nodes of the auxiliary network

3: $np *_t^{1..N_{nodes}}=0$; $W *_t^{1..N_{nodes}}=0$

4: for $j = 1$ to N_{nodes} do //project particles

5: $np *_t^j = np_{t-1}^j$

6: for $i = 1$ to nbc do

7: $\quad\quad \wp^{i}.\mathrm{np}_{t} * = \wp^{i}.\mathrm{np}_{t} * + \mathrm{np}_{t-1}^{j}$
8: $\quad$ end for
9: end for
10: $\Gamma_{t} = \Gamma_{t}^{*}$
11: for $j = 1$ to N_{nodes} do
12: $\quad$ if $\mathrm{np}_{t}^{j} \,!= 0$
13: //assign weight to the nodes with the likelihood observation function
14: $\quad\quad W_{t}^{j} = p(Y_{t}|X_{t}) * \mathrm{np}_{t}^{j}$
15: $\quad$ end if
16: end for
17: k=0;
18: for $j = 1$ to N_{nodes} do
19: $\quad\quad k = k + W_{t}^{j}$
20: end for
21: $Np = 0$
22: for $j = 1$ to N_{nodes} do
23: $\quad\quad W_{t}^{j} = W_{t}^{j}/k$ //normalize
24: $\quad\quad \mathrm{np}_{t}^{j} = \mathrm{round}\left(W_{t}^{j}/p_{min}\right)$ //resample
25: $\quad\quad Np = Np + \mathrm{np}_{t}^{j}$
26: end for
27: $Np_{remain} = 1/p_{min} - Np$
28: $i_{maxW} = \max_{i}(\omega_{t}^{i})$
29: $\mathrm{np}_{t}^{i_{maxW}} = \mathrm{np}_{t}^{i_{maxW}} + Np_{remain}$
30: $\mathrm{return}(\Gamma_{t}, i_{maxW})$

This particle filter without the resample function would suffer from a degenerative problem, this happens when after some iterations, all but one particle would show a weight near zero. The resample is an important step of this algorithm, it will force the particles to be distributed according to $p(X_{t}|Y_{t})$.

The (21) algorithm describes the implementation of the particle filter. In the first loop (lines 4-9) the particles are projected to adjacent nodes, according to the motion model defined by the sample frequency and the distance between nodes. The particles of each node have equal probability to move to the neighbor nodes, so the numbers of particles that exist in one node, are kept in the node and added to de adjacent nodes.

Then we have the prediction stage (lines 11-16) of the filter, it calculates the weight $p(Y_t|X_t)$ for each particle inside of the each node, so for the node n we have $W_t^n = p(Y_t|X_t) * np_t^n$. The weights W_t^n for each node are normalized (lines 18-26) based on the total weight K calculated adding all the W_t^i .

The resample is made with the use of the function $round(W_t^n/p_{min})$, this function transforms nodes with a weight superior to p_{min} into one or more particles, those particles that have a weight inferior to p_{min} simply disappear, from the nodes where they were. The particles that may have remained are added to the node with the highest W (lines 27-29) in this case the position of the target. Where $p_{min} = 1/\Omega$ and Ω is equal to a defined number of particles that should subsist in the network for the next iteration.

The function returns (Γ_t, i_{maxW}), the fingerprint set with new positions of the particles and the index of the node with the highest W, to be used as the position of the target, the target location is set to $T_t = \mathcal{L}_t^{i_{maxW}}$.

4.4. The measurement Model

The measurement model describes the process by which the likelihood observation function $p(Y_t|X_t)$ is generated. The likelihood observation function is generated having into account the measurement vector $Y_t = (\varrho_1, \dots, \varrho_{NumberOfAP})$ the state of the particles X_t^i and the fingerprinting vector or signature of each node $\mathcal{F}^n = (\rho_1, \dots, \rho_{NumberOfAP})$. The value of $p(Y_t|X_t)$ should decrease directly with the difference between the vectors fingerprint and measurement, $|Y_t - \mathcal{F}^n|$, where $NumberOfHeardAP \leq NumberOfAP$.

$$||Y_t - \mathcal{F}^n||^2 = \sum_1^{NumberOfHeardAP} (\varrho_i - \rho_i)^2 \qquad (23)$$

$$W_t^n = p(Y_t|X_t) = \frac{NumberOfHeardAP}{||Y_t - \mathcal{F}^n||^2} * np_t^n \qquad (24)$$

From the likelihood observation function $p(\mathbf{Y}_t|\mathbf{X}_t)$, of the particles, it is deduced the weight W for each node, taking the number of particles of the node (n) at t.

5. Experimental phase and results

5.1. Development

This phase consisted in developing two applications in java, which permitted to test the position algorithm written in C++. The first application is used to generate the network of nodes over a plant, running in a PC. The output of this application is a plain text file referencing the nodes and the connections.

The second application was developed to be used in a smartphone, with the OS Android, which permits the user to select the APs that will be used, with the location algorithm, and to construct the fingerprint data base. This application is composed by a graphical interface, where the area of test is represented by a bit map with the network of nodes and connections, represented by circles and lines, see figure 4.

To implement the fingerprint data base, the user stars by positioning over a node and sets an event to start the readings of the APs (this event consists in just touch the node where the user is), these readings take about 10 to 15 seconds and make at least 10 sample, from which is taken the average value for each AP representing the signature for a node. This procedure should be done for all the nodes that compose the network. The figure 4 a) is a screenshot of this application, where it is shown the nodes (in blue and red). The red means that the user didn't created the signature for that node and the blue ones have already a signature.

After the constructing of the fingerprint data base the application is ready to run the position algorithm. The user sets the algorithm by pressing the button "Next" and the application enter in the positioning mode. The figure 4 b) shows the position found by the algorithm, represented by a blue dot, ideally this dot should follow the position of the smartphone.

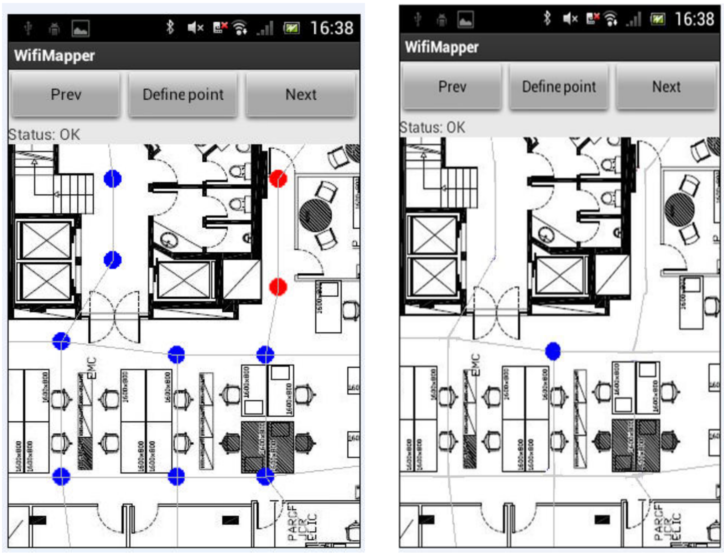

Figure 4. Screen shoots a) b)

5.2. Tests and results

For the test phase some constraints should be taken into account, regarding the filter and the environment. The input measurement vector Y_t is not the ideal, the body of the user forms a barrier to the radio waves and behaves as a source of noise. The other source of noise are persons that move around between the target and the APs, for a person that is positioned steady the measurement vector Y_t has noise associated that is not Gaussian.

The sample frequency of the measurement vector Y_t is not constant, the overcrowded number of Wi-Fi APs, make the readings of Y_t difficult and sometimes producing random a delays.

The variation of the sample frequency, the deterioration of the measurement vector Y_t and the velocity of the target may create some difficulties to the filter, it increases the error of the measured position and can make the filter to lose the position of the target, definitively. To overcome this last situation were created two instances of the algorithm. One instance is reissued every 10s (reset filter) and other is running continually (user filter). Having into account that the reset filter takes about 3 to 5 seconds to converge to the position of the target, after 5 seconds the reset filter target position is compared with the user filter target position. If the distance between the two is larger than a distance of one node, the state of the reset filter is copied to the user filter, with the new target position. Although this improves the performance of the filter, increases the processing load of the machine.

For the tests were studied two different situations, the static position and the dynamic position (it is considered “static position” a position that has more than 5sec). The performance of the static measurements are better than the dynamic measurements, in the dynamic measurements the algorithm presents a latency of about 1s or 2s representing a distance of 2 to 6 meters to be added to the static performance error.

The figure 5 is an histogram with the results of a test with the target statically positioned over a node and the adjacent nodes at 2,3 and 5 meters. The testes were made with 3, 4 and 5 APs, a number of less or equal to 2 AP were not used because the accuracy was shown be too poor for the applications requirements.

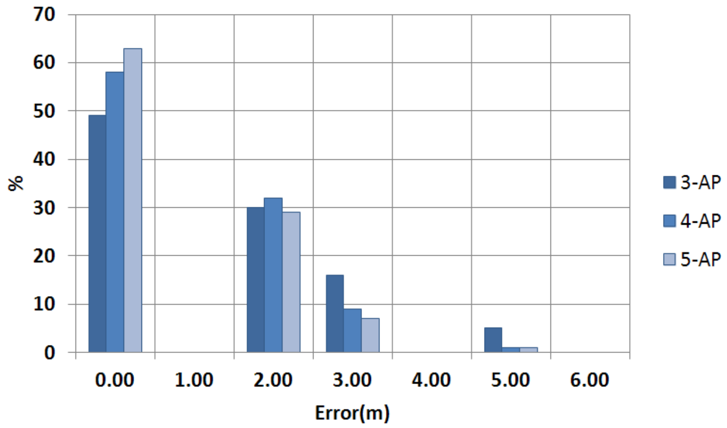

Figure 5. Histogram of the target positioned over a node

The figure 6 is a histogram with the target, statically positioned, at a distance of 2,3 and 5 meters from nodes 3 nodes.

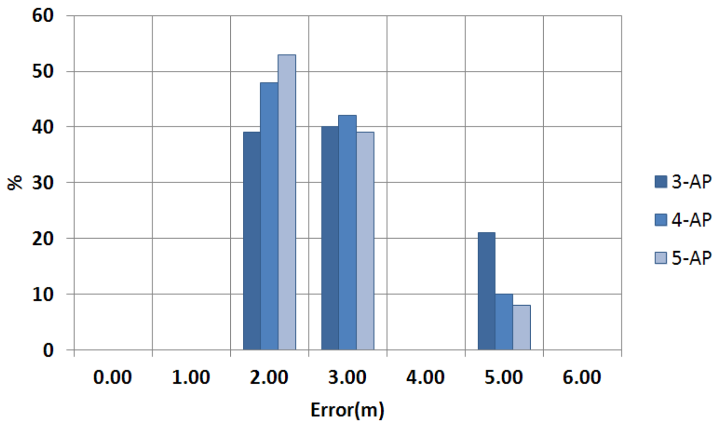

Figure 6. Histogram of the target positioned at 2 meters from a node

These resuslts are only indicative because the environment is critical to the performance of this kind of positioning. In small areas defined by brick walls like areas A and B, from figure 7, the performance of the algorithm is considerably better than in open space similar to area C. This happens because the small areas confined by brick walls produce signatures well defined and in open space the signatures are not so well defined, two nodes may have very similar signatures resulting in W_t^n almost equal for adjacent the nodes.

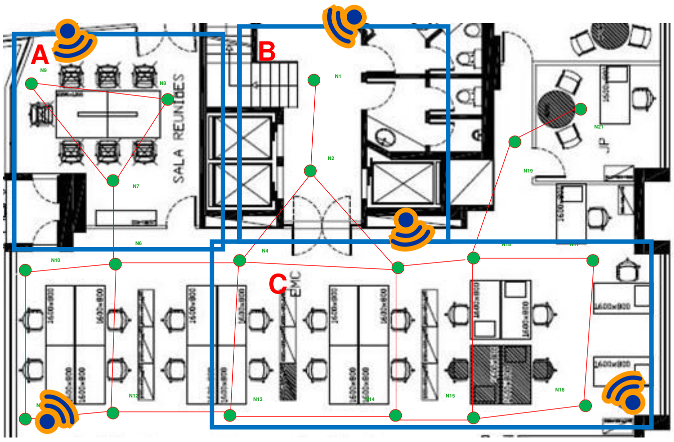

Figure 7. Test set-up with 5 APs

The distribution of the APs was also important for the accuracy of the system, it was verified that the APs should be placed around test the area. If the APs are placed internally to the area, mainly at the center or aligned, results in a poor performance.

6. Conclusions

The present approach presents an alternative to a particle filter that does not dependent on the number of particles but only on the number of nodes, improving the use of computation consumption. For the, measured position, were not used statistical algorithms to interpolate positions between the nodes, which would heavier the process. Taking into account the results of the work [8], which showed that similar positioning systems would have, at the best , an accuracy of 3 meters (for 80% of the samples), for that it is no use to represent an intermediate position. Having in consideration the required accuracy for the hospital and museum application the method shows a good performance.

References

[1] Dr. Widyawan "Learning Data Fusion for Indoor Localisation", Cork Institute of Technology, October 2009

[2] Dr. Rainer Mautz, "Indoor Positioning Technologies", ETH Zurich, February 2012

[3] Dr. Nattapong Swangmuang "A location Fingerprint Framework Towards Efficient Wireless Indoor Positioning Systems", University of Pittsburgh, 2008

[4] V. Honkavirta, T. Perala, S. Ali-Loytty, and R. Piche, "A Comparative Survey of Wlan Location Fingerprinting Methods," Proc. Sixth Workshop Positioning, Navigation and Comm., pp. 243-251, 2009.

[5] H. Liu, H. Darabi, P. Banerjee, J. Liu; "Survey of Wireless Indoor Positioning Techniques and Systems", IEEE Transactions on Systems, November 2007

[6] Faria D. B., "Modeling Signal Attenuation in IEEE 802.11 Wireless LANs - Vol. 1", Technical Report TR-KP06-0118, Stanford University,2005

[7] Arulampalam, M.S., Maskell, S., Gordon, N. & Clapp, T. "A tutorial on particle filters for online nonlinear/non-gaussian Bayesian tracking.", IEEE Transactions on Signal Processing. 50(2):174-188

[8] Y. Zhao, Q. Shen, and L. Zhang, "A novel high accuracy indoor positioning system based on wireless lans," Progress In Electromagnetics Research C, Vol. 24, 25-42, 2011.