

# Analysis of the Behavior of an Indoor Position System Based on Fingerprints and IEEE 802.15.4

A. Verónica Medina<sup>\*1</sup>, José A. Gómez<sup>2</sup>, José A. Ribeiro<sup>3</sup>, Enrique Dorrnoro<sup>4</sup>, Sergio Martín<sup>5</sup>

Departamento Tecnología Electrónica, Seville University, TAIS Group (<http://matrix.dte.us.es/grupotais/>)  
ETSI-INF, Avda. Reina Mercedes s/n, Seville, Spain

<sup>\*1</sup>vmedina@us.es; <sup>2</sup>jpgomez@dte.us.es; <sup>3</sup>jose.ribeirolimpo@gmail.com; <sup>4</sup>enriquez@dte.us.es; <sup>5</sup>smartin@us.es

**Abstract-** This paper presents an analysis of the behaviour of an indoor position system based on fingerprints and IEEE 802.15.4 that has been adapted to be tested in a competition, called EvAAL (Evaluating AAL Systems through Competitive Benchmarking), held both in Madrid (tests) and Eindhoven (results). The objectives of this analysis are to determine the best algorithm that should have been applied in order to obtain the best results in the competition to use them in other environments. Among the different combinations that can be applied, i.e., the way the signature database is filled in and the algorithm uses to determine the closest location point, the best results are obtained using a global signature database where each signature entry is calculated by the medium of samples signatures database and the closest location is determined by a centroid algorithm with the parameter  $c$  set to 1.3. In this way, the error made improves the one obtained in the EvAAL, which is reduced by 100 centimetres.

**Keywords-** IEEE 802.15.4; RSSI; Centroid; Indoor position; ZigBee; WSN; BitCloud; OpenMAC

## I. INTRODUCTION

WSNs (Wireless Sensor Networks) are presented in many applications, and examples of WSN applications are found in Ambient Living [1-4] or Smart building [5-9] researching fields for solving data acquisition process. Depending on its applications, ambient or user sensors and actuators can be used for making decisions. WSNs are composed by motes. A mote is a device that conforms to the IEEE 802.15.4 standard, so a WSN has this technology as based.

The knowledge of a subject's position is very useful in these kinds of systems because depending on it the decisions to be made are different. As stated in [11] and [12], an amount of indoor location tracking systems have been proposed in the literature, based on Radio Frequency (RF) signals, ultrasound, infrared, or some combination of modalities.

Using RF signal strength, it is possible to determine the location of a mobile node with an acceptable accuracy. Given a model of radio signal propagation in a building or other environment, received signal strength can be used to estimate the distance from a transmitter to a receiver, and thereby to triangulate the position of a mobile node. However, this approach requires detailed models of RF propagation and does not account for variations in receiver sensitivity and orientation.

An alternative approach is to use empirical measurements of received radio signals, known as RSSI, Receiver Signal Strength Indicator, to estimate location. By recording a database of radio "signatures" along with their known locations, a mobile node position can be estimated by acquiring the actual signature and comparing it to the known signatures in the database, also known as fingerprints. A weighting scheme can be used to estimate location when multiple signatures are close to the acquired signature.

All of these systems require the signature database to be manually collected prior to system installation, and rely on a central server (or the user's mobile node) to perform the location calculation. Several systems have demonstrated the viability of this approach, one of those is MoteTrack [11-12]. MoteTrack is based on deploying a specific WSN to determine location.

MoteTrack's basic location estimation uses a signature based approach that is largely similar to RADAR [10] that obtains a 75th percentile location error of just under 5 m, but in MoteTrack decreased the location error by 1/3.

We have implemented a similar system to MoteTrack, a signature-based localization scheme, but using other motes and different software, the BitCloud Stack [13], a Zigbee and Zigbee-PRO implementation, and OpenMac Stack [14], an IEEE 802.15.4 implementation. The way the messages are sent and how the RSSI is calculated is also different as the one used in MoteTrack. We have also compared both implementations in [15] and presented in [16] an adaptation of our system in the EvAAL competition.

In this paper, an analysis of the behaviour of the adapted indoor positioning system is presented in order to determine the best algorithm to be applied in an environment similar to the one tested in the EvAAL.

This paper is structured in the following way. An overview of our indoor positioning prototype and the adaptation made for the EvAAL competition are shown in Sections II and III. In Section IV the methodology followed to gather all the information to be analysed is explained. The simulations made are presented in Section V. Finally conclusions are established in Section VI.

II. PROTOTYPE OVERVIEW

In our prototype (Fig. 1), a building or other area is populated with a number of motes acting as fixed nodes, one of them being the coordinator, *C*, and a set of motes as mobile nodes, the ones whose position is going to be determined. Each fixed node sends to *C* periodic beacon messages, beacon 2, which consist of an *n*-tuple of the format {MobileID, RSSI}, where *n* is the number of mobile nodes, MobileID is a unique identifier of a mobile node, and RSSI is the signal strength which each fixed node received the last beacon message sent by MobileID node. The beacon message sent by a mobile node is different from the one sent by a fixed node, to differ one from others, the mobile node beacon messages are called beacon 1. Not all fixed motes receive beacon 1 messages, and this depends on the coverage area. In this case they send a beacon 2 with a zero value in RSSI corresponding to that mobile node.

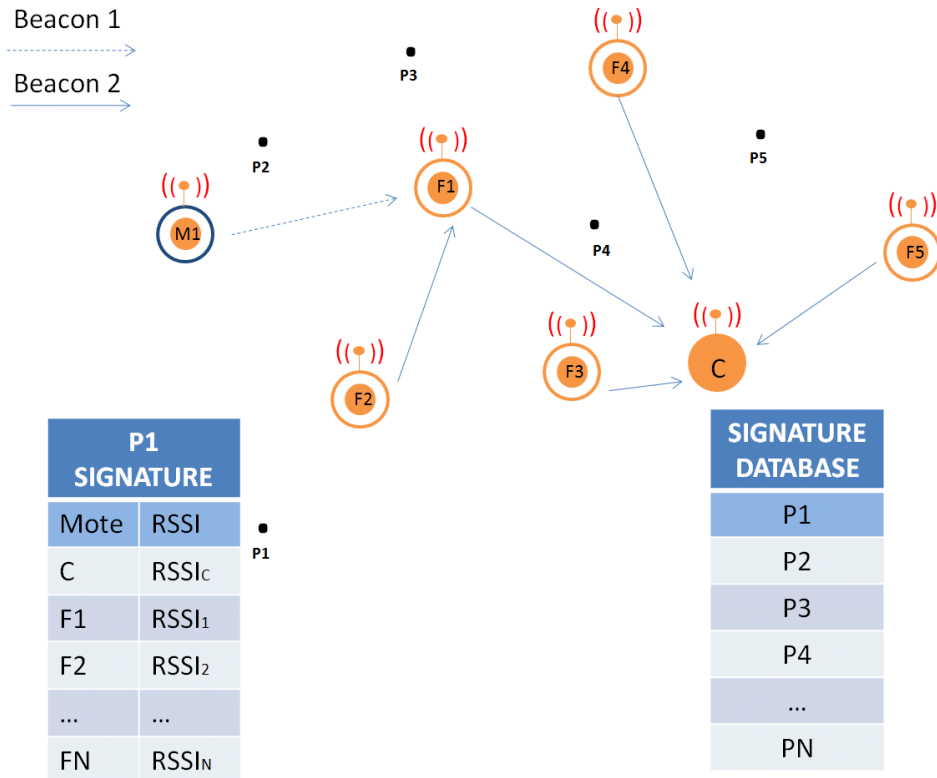


Fig. 1 System Overview. M1 is a mobile node, F1-F5 are fixed nodes, and C is the coordinator, also a fixed node. M1 periodically sends a beacon message, beacon 1, to inform the others node that is present, all fixed node that receives it, save the RSSI of that message in a table. Fixed node periodically sends a message to C, beacon 2, to inform about the RSSI that they receive from mobiles node, M1 in this case.

The location estimation problem consists of a two-phase process: an offline collection of reference signatures to set the signature database, followed by an online location where the mobile nodes position are estimated.

A. Offline Phase

As in other signature-based systems, the reference signature database, *R*, in the off-line phase, is acquired manually by a user with a mobile node and a PC connected to *C*. The reference signature database consists of a number of reference signatures. Each reference signature, shown as black dots in Fig. 1, is formed by a set of signature tuples of the format {source ID, mean RSSI}, where source ID is the fixed node ID and mean RSSI is the mean RSSI of a set of beacon messages received over some time interval. The mean RSSI is used because it is the way MoteTrack did, but it is possible to use another criteria. Each signature is mapped to a known location by the user acquiring the signature database (P1-P5 in Fig. 1).

B. Online Phase

In the online phase, given a mobile node's received signature, *s*, received from the fixed nodes, and the reference signature set *R*, the mobile node's location can be estimated in the following way. The first step is to compute the signature distances, from *s* to each reference signature  $ri \in R$ . We employ the Manhattan distance metric, as were done in MoteTrack,

$$M(r, s) = \sum_{t \in T} |RSSI(t)r - RSSI(t)s| \tag{1}$$

where *T* is the set of signatures tuples presented in both signatures,  $RSSI(i)r$  is the RSSI value in the signature appearing in signature *ri* and  $RSSI(i)s$  is the RSSI value in the signature appearing in signature *s*.

Given the set of signature distances, the location of a mobile node can be calculated in several ways applying the Centroid algorithm

Centroid algorithm considers the centroid of the set of signatures within some ratio of the nearest reference signature. Given a signature  $s$ , a set of reference signatures  $R$ , and the nearest signature we select all reference signatures  $r \in R$  that satisfy

$$\frac{M(r, s)}{M(r^*, s)} \tag{2}$$

for some constant  $c$ , empirically-determined. The geographic centroid of the locations of this subset of reference signatures is then taken as the mobile node’s position. Small values of  $c$  work well, generally 1.1 or 1.2. If  $c=1$  the position estimation is the position of the nearest signature saved in the signature database. We used in our prototype  $c=1.2$  as in MoteTrack.

C. Prototype Uses and Tests

This prototype was used in a research project that tried to make an Intelligent Building<sup>1</sup>. The Building had to adapt the environment to make its users feel comfortable by controlling air-conditioning, music, etc., by means of the WSN deployed in it that senses the environment. The users of the building had to carry a mote (the mobile mote) and the decision maker software informed the actuator software to change the environmental devices as user requirements by using the output of our system (estimated position and sensors information) and other parameters they estimated. The focus of that research project was only an accuracy of room positioning.

The prototype (Fig. 2) was deployed over half floor of our Department Area, measuring roughly 225 m<sup>2</sup>. After testing how the different kinds of materials affect the RSSI value and that a mote can cover an area of 4-5 meters, we determined that a number of 7 fixed motes were enough to cover the whole area. Our prototype was tested in order to know if it is possible to determine if a mobile mote is placed in a room, i.e., it did not matter exactly where it was inside the room, so the precision required was not very high. This was this way, because the kind of applications for whom our indoor position solution was tested did not require more precision.

Based on empirical measurements, we determined that the precision of our prototype was about 77%, i.e., the right room was determined in that percentage being the accuracy among 0 meter to less than 1 meter from the real position. The rest one was bad position determination, not the right room, but the accuracy was among 0.5 meters to less than 4 meters from

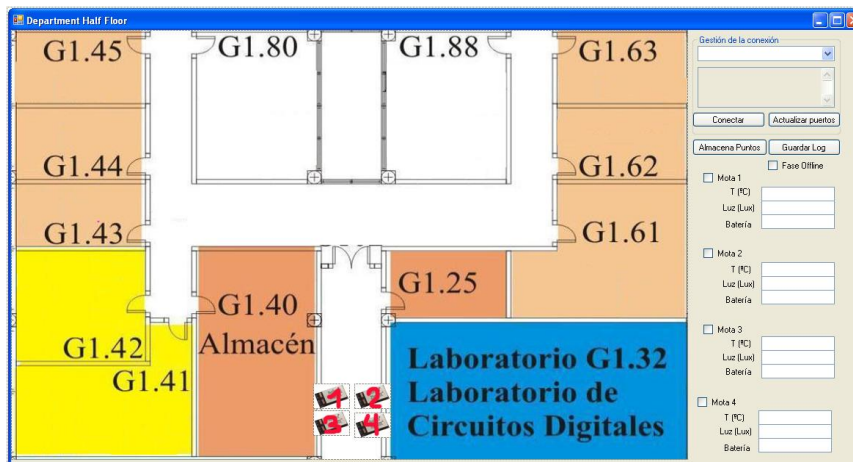


Fig. 2 Prototype interface

III. ADAPTATION TO THE EVAAL COMPETITION

As mentioned previously, the design requirements of our prototype were only to determine the actual position in a room of a user in a building, so the prototype accuracy was room accuracy. In spite of the fact that one of the most important drawbacks was that the Smart House Living Lab of the Polytechnic University of Madrid had only two rooms we decided to compete in the second edition of EvaAL Competition. This implied that the required accuracy was meters (error less than or equals to 0.5 meters got the higher score, higher than 4 meters got no score) and the room accuracy was substituted by areas of interest (AOI), so the behaviour of how our prototype was going to work was an incognita.

<sup>1</sup> Health Intelligent Technologies Oriented to Health and comfort in Interior Environments (TECNO-CAI) approved project at the fifth call of CENIT program by the Innovation Science Ministry of Spain (CDTI and Ingenio 2010 Program).

Among other things, we had to adapt the user interface (Fig. 3) to the new area and floor plant. The area was approximately 100 m<sup>2</sup>. We also had to decide the numbers of motes required and the place where each one was going to be placed. The prototype also had to interact with a domotic bus deployed in the Living Lab, and the benchmark software. Although the domotic bus events were received in the prototype, such as a light switch when it was switched on, to name one of them, they were not taken into account in determining the actual position.

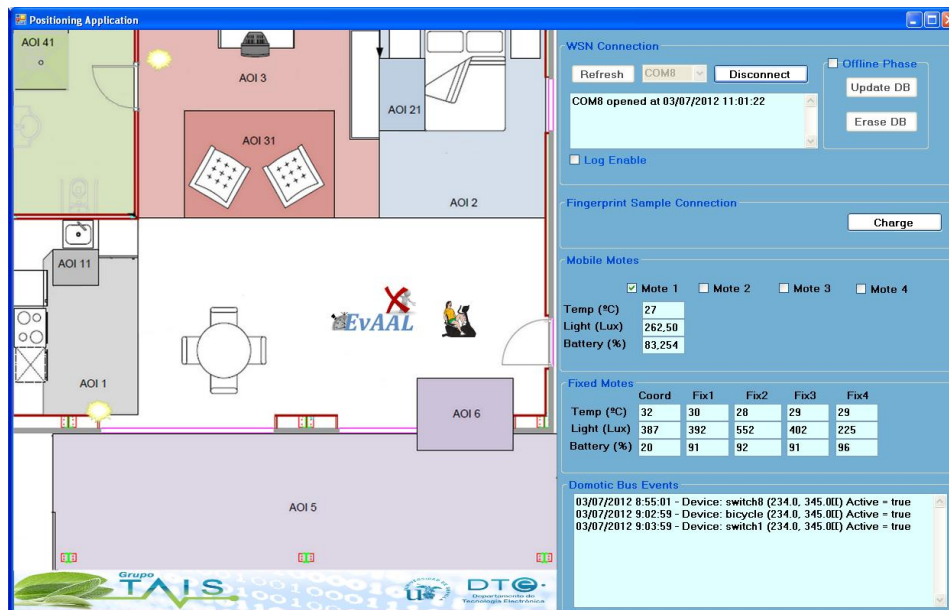


Fig. 3 EvAAL prototype interface

The competition results are shown in Table 1. There were five main topics to be scored: accuracy, availability, installation complexity, user acceptance and AAL environment integration. Accuracy measured the position error in meters in 75th percentile and AOI success, in our case, we only got point in AOI success. Availability tested how often the prototypes sent measurements, we got the higher score in this item among all competitors. Installation complexity tried to evaluate how easy the prototype could be installed, due to the rush, we did not pay attention to this item as it is going to be explained in the next paragraph. User acceptance focused on how comfortable the prototype was from user's point of view and AAL environment integration checked how easy the prototype could be integrated in AAL environment (for instance, if it got standard interfaces). We got a final score of 4.2 points.

TABLE 1 EVAAL COMPETITION RESULTS

	Accuracy	Availability	Installation Complexity	User Acceptance	AAL Environment Integration
Weight	0,25	0,2	0,15	0,25	0,15
Score	0,67180617	10	0	5,11217949	5,15476191
Total:	0,16795154	2	0	1,27804487	0,77321429
	Final Score:		4,2192107		

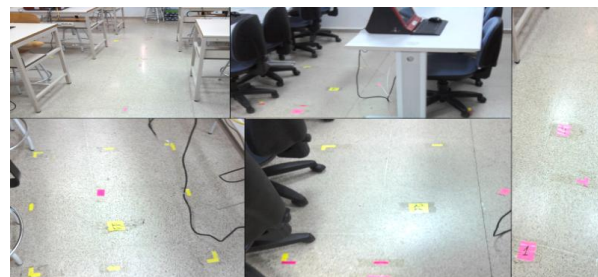


Fig. 4 Living Lab emulation photos



Fig. 5 Living Lab emulation plant view

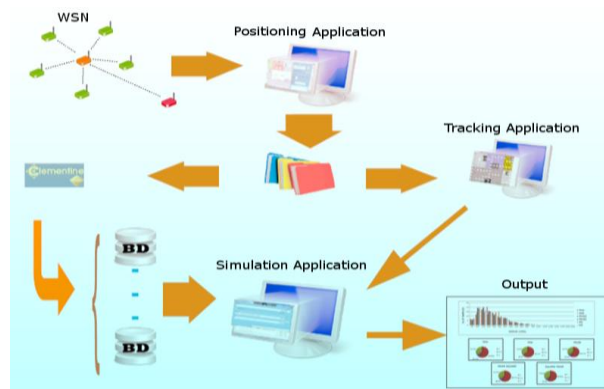


Fig. 6 Testing methodology

All the competitors knew the evaluation criteria before the test day in Living Lab. In our case, we did not pay attention because we focused on adapting our prototype to the Living Lab. We did not even have time left to test correctly the adapted prototype prepared to the EvAAL. For a quick testing, we deployed the Living Lab in two of our laboratories, marking on the floor the different rooms and AOIs as shown in Fig. 4. The emulation was not quite good due to the fact that there was a wall between the kitchen and the living-room zone (Fig. 5).

Our competition results were not very good but they were worse than we expected. In our previous and quick test, without a deep study, the point to point accuracy was 3, 1 meters in 75th percentile, but the result in Living Lab was 4.6 meters. The AOI accuracy result in our labs was 59% being 40% the right AOI and 19% not the right AOI but a subarea in that AOI. In Madrid we got a 20% in AOI accuracy which the right AOI score was 5.5%. So the wall in the labs we tested the adapted prototype should have made the emulation not as real as we thought.

#### IV. METHODOLOGY

The fingerprint algorithm we use to determine the position has many parameters that can be set depending on the environment it is going to work so the accuracy results could be different if they are not set adequately. These parameters are, among others, the way each signature point is calculated, how samples are gathered, how the distance between two signatures is measured, etc. We tested our prototype in the Living Lab in Madrid without studying deeply the right combination for them in that environment. In this section we explain the methodology followed to gather the data in order to make simulations to test which parameters are the most appropriate, to achieve the best results, i.e., to improve the results obtained in the EvAAL competition.

The whole methodology applied is shown in Figure 6. First we got from the WSN, using the prototype software, for each point in the signatures database a set of samples, samples database, in order to determine later its signature by using. Each sample saved had the following information: year, month, day, hour, minute, second, x-coordinate, y-coordinate, orientation, where orientation can refer to north, south, east, west, or global. If orientation is global it means that the sample was saved while the mobile mote was turning around. In this way, different signatures database can be calculated according to diverse criteria.

For tracking purpose, some extra points were taken, in this case, the orientation was not global. A tracking application was also developed to define different tracks to test.

Different signature databases were calculated with the sample database points using a data mining tool called Clementine.

A simulation program was also developed to try how different algorithms behave using the same database and track. The output of this program for different scenarios let us know which one would be the best combination.

#### V. SIMULATIONS

Several tests have been made. For each one, a specific database, track and algorithm were tried. The most significant change among different simulation tests is how the signature database points were calculated. Firstly, two kinds of signature database were possible, one global and the other directional (north, west, south, east). Secondly, each point in the signature database was set calculating the mean, mode, max, min, etc, of all the sample points acquired previously.

As presented in Section II, in the off-line phase the acquired signature is compared with the ones saved in the signature database applying the Manhattan distance to determine the closest signature points (one or more depending on  $c$  parameter) in the signature database. In the EvAAL competition, the Euclidian distance was used to measure the error made, so we have also tried this distance, both to compare a signature with the signatures database and to measure the accuracy error. The actual position is determined using the centroid algorithm. This algorithm has the  $c$  parameter. Depending on it, the results were different. We tried from 1 to 1.7, higher values had worse results.

As one can image, the amount of information obtained to be analysed and compared was too much. For each signature database obtained you had to simulate the diverse track to be analysed and  $c$  parameters to be tested, so there were hundreds of combinations. Manage all these results appropriately, we got the next conclusions.

#### VI. CONCLUSIONS

Fig. 7 and Fig. 8 shows respectively for a parameter  $c=1.3$  the results obtained using the same track with a signature database global or the signature database directional. The above graphic in Figs. 7 and 8 represents the relationship between the sample dispersion in percentage and the error made in centimetres. This shows that the election of a specific database together with a good coefficient  $c$  might improve the results. The graphics below represents the success in AOI. The score was similar to the EvAAL competition. 0 failed, 1 success, and 0.5 failed but it was in a subarea in the AOI. The conclusions are the same, changing parameters might improve results.

In order to compare adequately the different combinations and to determine the best, we decided to use the error made in the 75th percentile, i.e., the same criteria used in EvAAL competition. In Fig. 9 it is shown among all the tests made, the one



which error was below 365 centimetres and in Fig. 10, it is shown the success in AOI for the best results.

The best results are obtained using a global signature database, where the points in it are calculated using the medium of the squares of samples database with  $c=1.3$  and the medium of samples database with  $c=1.4$ . The error is respectively, 360 cm and 361 cm. Although in the first case, the error is less, it does not imply a better result in the AOI success as shown in Figure 10. That is not wrong because one calculated coordinate could be closer to the real one but that coordinate belongs to another AOI. So it can be concluded that using a global database using the medium of samples with  $c=1.3$ , the results obtained in the EvAAL competition are improved in one meter.

Finally, we realized that we did not measure correctly the error made by our prototype and the first Living Lab emulation we made when we tested it before the competition day, that is why our score in accuracy was worse than we expected.

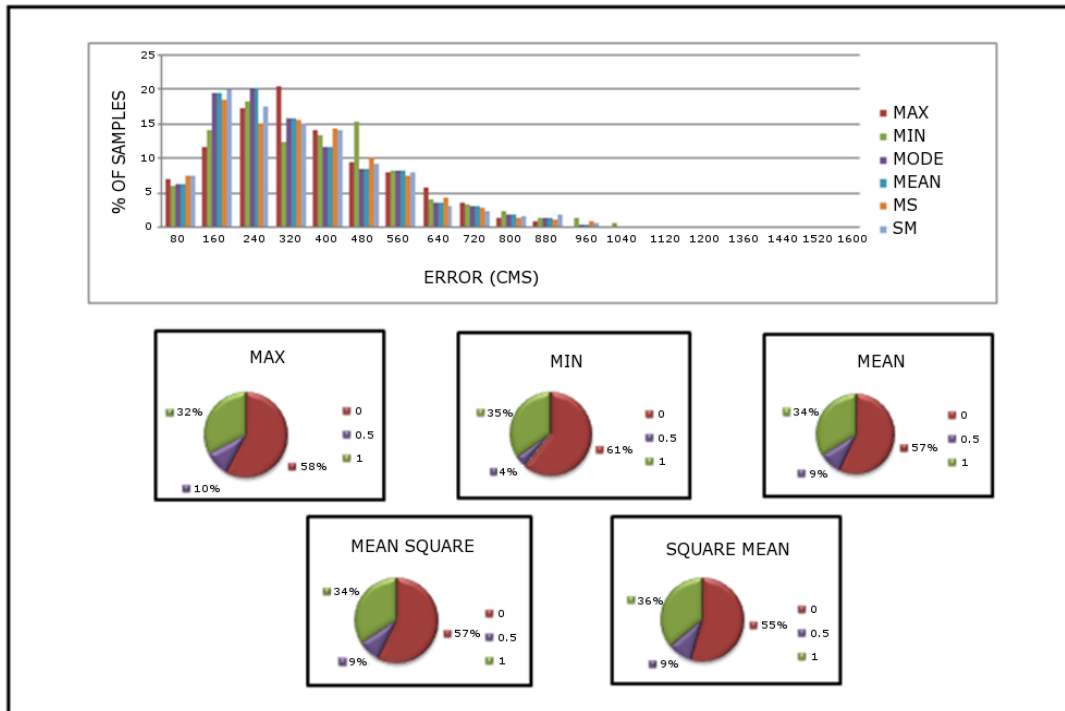


Fig. 7 C=1.3-directional signature database

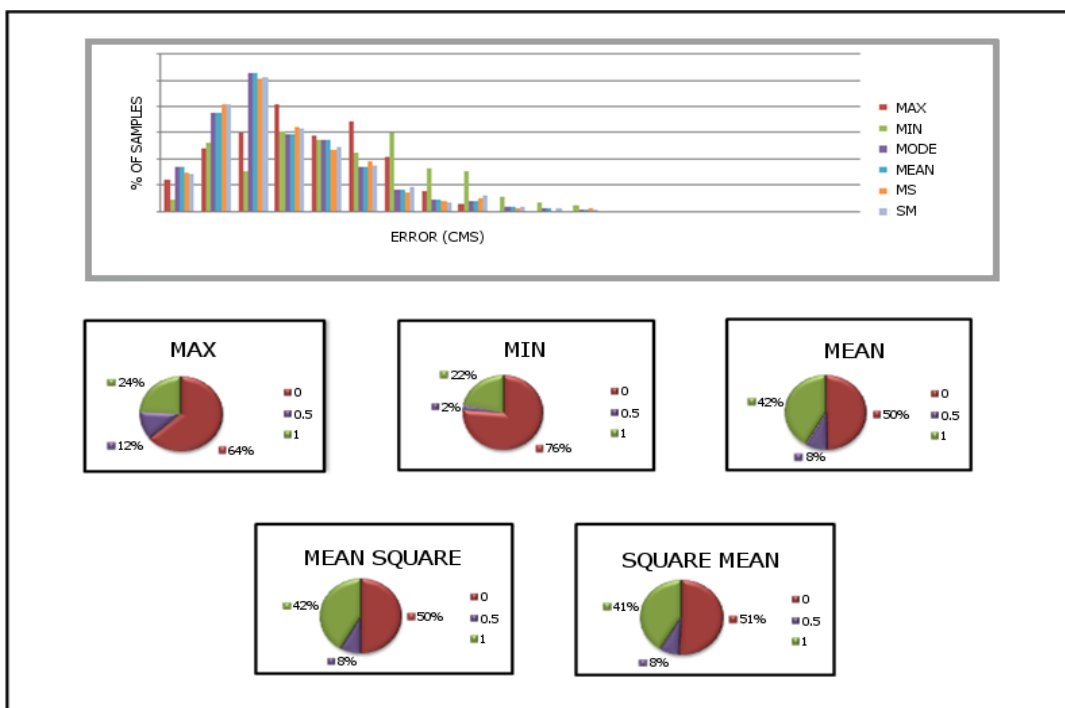


Fig. 8 C=1.3-global signature database

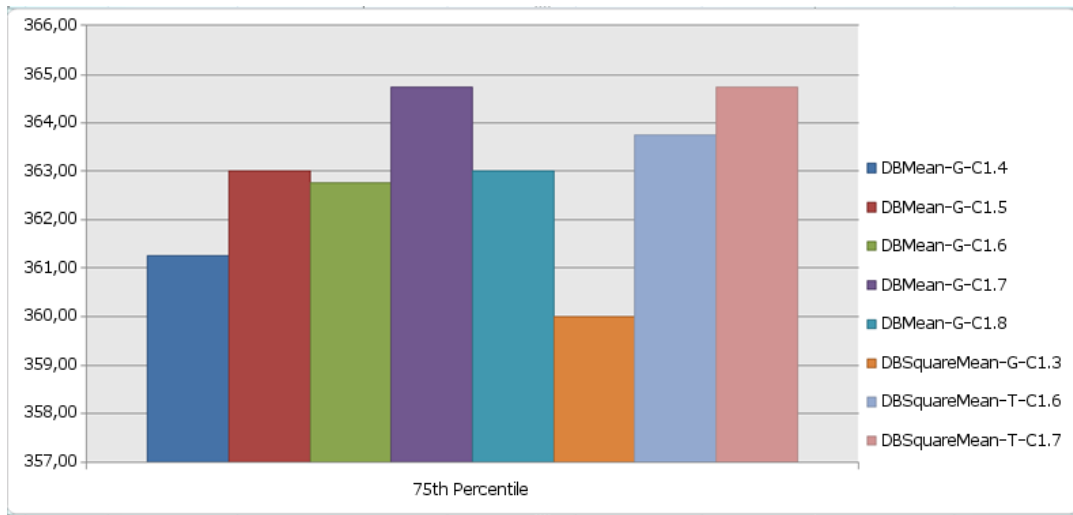


Fig. 9 Error made in percentile 75

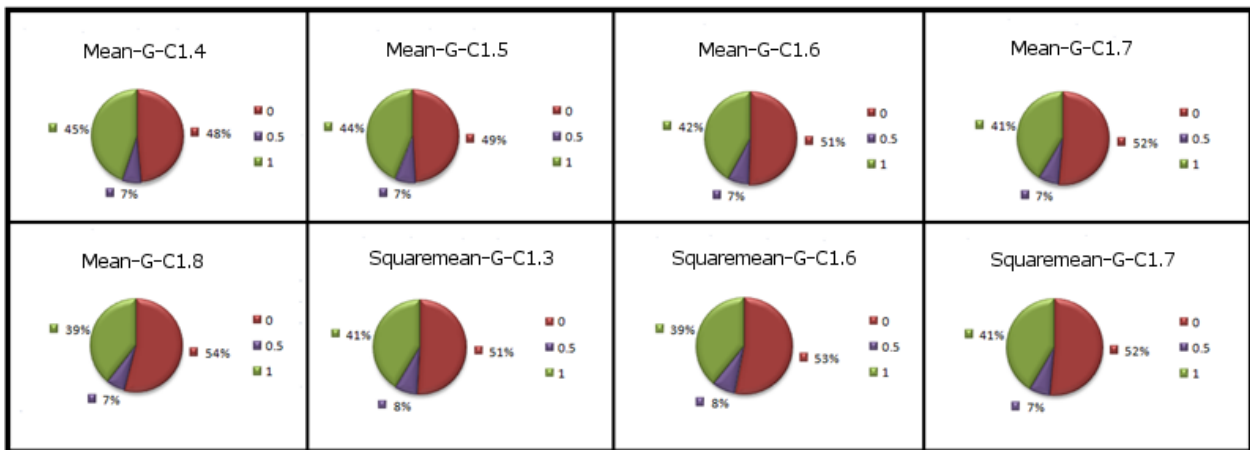


Fig. 10 AOI accuracy

VII. ACKNOWLEDGMENT

This work has been carried out within the framework of two research programs: (P08-TIC-3631) – Multimodal Wireless interface (IMI) funded by the Regional Government of Andalusia.

REFERENCES

- [1] Hristova, A., Bernardos, A.M., and Casar, J.R. Context-aware services for ambient assisted living: A case-study. First International Symposium on Applied Sciences on Biomedical and Communication Technologies, 2008. ISABEL '08.
- [2] Figueiredo, C.P., Gama, O.S., Pereira, C.M., Mendes, P.M., Silva, S., Domingues, L., and Hoffmann, K.P. 2010. Autonomy Suitability of Wireless Modules for Ambient Assisted Living Applications: WiFi, ZigBee, and Proprietary Devices. *Sensor Technologies and Applications (SENSORCOMM)*.
- [3] Hong Sun, De Florio, V., Ning Gui, and Blondia, C., Towards. 2008. Building Virtual Community for Ambient Assisted Living. 16<sup>th</sup> Euromicro Conference on Parallel, Distributed and Network-Based Processing.
- [4] Sun, H., De Florio, V., Gui, N., and Blondia, C., PRomises and Challenges of Ambient Assisted Living Systems. *Information Technology: New Generations, 2009. ITNG '09. Sixth International Conference on*. pp. 1201-1207, 2009.
- [5] Martin, H., Bernardos, A.M., Bergesio, L., and Tarrio, P., Analysis of key aspects to manage wireless sensor networks in ambient assisted living environments. *Applied Sciences in Biomedical and Communication Technologies, 2009. ISABEL 2009. 2<sup>nd</sup> International Symposium*. pp. 1-8, 2009.
- [6] Dietrich, D., Bruckner, D., Zucker, G., and Palensky, P., Communication and Computation in Buildings: A Short Introduction and Overview. *IEEE Transaction on Industrial Electronic*, vol. 57, iss. 11, 2010.
- [7] Chen, Po-Wei, Ou, Kuang-Shun, and Chen, Kuo-Shen., IR indoor localization and wireless transmission for motion control in smart building applications based on Wiimote technology. *SICE Annual Conference 2010*.
- [8] Han Chen, Chou, P., Duri, S., Hui Lei, and Reason, J. The Design and Implementation of a Smart Building Control. pp. 255-262, 2009.
- [9] Snoonian, D., Smart buildings. *Spectrum, IEEE* vol. 40, iss. 8, Digital Object Identifier: 10.1109/MSPEC. pp. 18- 23, 2003.
- [10] Bahl P., and Padmanabhan VN. RADAR: an in-building RF-based user location and tracking system. In: *INFOCOM*, pp. 775-784, 2000.

- [11] Konrad L. and Matt W. MoteTrack: A Robust, Decentralized Approach to RF-Based Location Tracking. In Proceedings of the International Workshop on Location and Context-Awareness (LoCA 2005) at Pervasive May 2005.
- [12] Konrad L., Matt W. MoteTrack: A Robust, Decentralized Approach to RF-Based Location Tracking. To Appear in Springer Personal and Ubiquitous Computing, Special Issue on Location and Context-Awareness. ISSN: 1617-4909 (Print) 1617-4917, 2006.
- [13] Medina A. V., Gómez I., Romera M., Gómez J.A. and Dorrnoro E. 2011. Indoor Position System based on BitCloud Stack for Ambient Living and Smart Buildings. In 3<sup>rd</sup> International ICST Conference on IT Revolutions. Córdoba, Spain.
- [14] Medina A.V., Gómez J.A., Rivera O., Dorrnoro E., and Merino M. 2011. Fingerprint Indoor Position System based on OpenMAC. In International Conference on Wireless Information Networks and Systems, Sevilla Spain.
- [15] José Antonio Gómez Martín, Ana Verónica Medina Rodríguez, Enrique Dorrnoro Zubiete, Octavio Rivera Romero, Sergio Martín Guillén. 2013. Journal of Networks, vol. 8, iss. 1, pp. 37-44, Jan 2013. DOI:10.4304/jnw.8.1.37-44. ISSN 1796-2056.
- [16] A. Verónica Medina, José A. Gómez, José A. Ribeiro, Enrique Dorrnoro. 2013. Indoor Position System Based on a Zigbee Network. Evaluating AAL Systems Through Competitive Benchmarking. Communications in Computer and Information Science, vol. 362, 2013, pp. 6-16. ISSN: 1865-0929. DOI: 10.1007/978-3-642-37419-7\_2.