2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 4-7 October 2016, Alcalá de Henares, Spain

Indoor tracking from multidimensional sensor data

UMINHO at the IPIN 2016 Competition

Adriano Moreira, Maria João Nicolau, António Costa
Algoritmi Research Centre
University of Minho
Guimarães, Portugal
{adriano.moreira, maria.j.nicolau, antonio.costa}
@algoritmi.uminho.pt

Filipe Meneses
Centro de Computação Gráfica (CCG)
Guimarães, Portugal
Filipe.Meneses@ccg.pt

Abstract— **Tracking the position of people or vehicles in large indoor settings with high accuracy is still a challenge despite the significant progress observed in indoor positioning technology in the last decade. To date, there is not a clearly dominant indoor positioning solution for general use, and challenges related to seamless indoor-outdoor positioning, reliable floor estimation and indoor maps are still needing more research. In this context, the IPIN 2016 conference is promoting a competition to evaluate a set of competing indoor positioning solutions in a realistic scenario. This paper describes the proposal of the UMINHO team and some of the obtained results.**

Keywords—indoor positioning; competition; tracking; WiFi fingerprinting; floor detection

I. INTRODUCTION

The IPIN 2016 Indoor Localization Competition follows previous editions of similar competitions organized by the EvALL project [1] and hosted by the IPIN conference since 2014 [2,3]. As in the two previous editions, this competition includes an "off-site" track, where competitors develop and test their positioning estimation algorithms using a set of data provided by the competition organizers. In 2015, the "off-site" track focused on WiFi fingerprinting [4], and the provided data included training and validation datasets made of WiFi fingerprints collected using several smartphones [2,5]. The competing teams could use these datasets to build radio maps and to tune their algorithms. A set of evaluation log files was eventually provided to the competitors without ground truth, containing a set of fingerprints for which the teams had to estimate a position (building, floor and pair of coordinates).

The corresponding track at the IPIN 2016 Competition, named "Smartphone-based (off-site)", while inheriting some of the characteristics from its 2015 counterpart, is significantly more challenging [6]. It opens, however, the opportunity to explore more advanced positioning estimation algorithms.

This paper describes how the UMINHO team addressed this competition. Section II briefly introduces the competition, with its objectives and rules, and describes the data provided to the competing teams. In section III we describe how the provided data was processed to build a radio map to support our WiFi fingerprinting-based position estimation process. Section IV describes the fundamentals of our positioning estimation method and the process used to generate the estimated trajectories from the provided evaluation data. In section V some results are presented, even without any performance metric since this paper was written before the final assessment of the competition results. In section VI we draw some conclusions from this work and discuss some ideas for future competitions.

978-1-5090-2425-4/16/$31.00 ©2016 IEEE

II. THE SMARTPHONE-BASED (OFF-SITE) COMPETITION

One of the features of this year's datasets is that all the data records where collected sequentially in time and its many dimensions are time synchronized. Datasets are organized in log files, one per trajectory, collected using different smartphones carried by a walking human.

Data for this competition was collected in three different cities, in four different buildings (two of the buildings are at the same university campus), and across 1 to 6 floors per building. Data with ground truth is organized in datasets, each one supplied as a single CSV file. There are a total of 17 datasets corresponding to 10 distinct routes. Some of these routes also include outdoors trajectories.

The provided datasets are made of a set of records, all collected using a specially crafted Android app, and referring to a set of sensors, including: WiFi fingerprints (SSID, MAC address of the Access Point, RSSI); magnetic (3 axes); accelerometer (3 axes); gyroscope (3 axes); atmospheric pressure; light intensity; sound level; temperature; humidity; mobile phone orientation (pitch, roll and yaw); and satellite-based position (latitude, longitude, bearing, accuracy and speed). Additionally, there are some records with ground-truth referring to a set of reference points visited during the continuous data collection process (building, floor, latitude, and longitude). All the records are labelled with one or two timestamps (sensor and phone timestamps).

A. Time synchronization and radio maps

Although all data records are time synchronized, data obtained from different sensors were collected at different rates. Moreover, the ground truth records (POSI) were not collected at the exactly the same time instants as the WiFi fingerprints (WIFI), which results in a higher difficulty in building a good quality radio map. Fig. 1 illustrates this issue. As each WiFi sample does not have a clear ground truth sample associated with it, a method to estimate the position to each WiFi sample is required. The same applies to samples collected from other

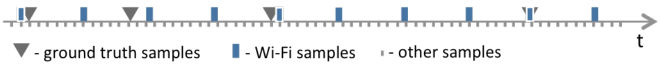

Fig. 1 Samples from different sensors.

sensors. How to estimate these positions depends on the sampling rate associated to each sensor.

While WiFi records were collected with an average sampling period of 4.7 seconds, POSI records were collected across a much wider set of sampling periods, with an average of 9.6 seconds (up to a maximum of 77.4 seconds between consecutive records). This means that there are around only one POSI sample for each two consecutive WiFi samples, on average. The time difference between a WiFi sample and the nearest (in time) POSI sample is also variable, with a mean of 4.8 seconds and a maximum of 38.7 seconds. Moreover, data were collected with the user walking at different speeds, as observed through the relationship between the displacement and elapsed time between consecutive POSI samples (Fig. 2).

Data from other sources were collected at different sampling rates. Relevant for our approach, atmospheric pressure (PRES) and accelerometer (ACCE) data records were collected with average sampling periods of 0.05 and 0.02 seconds, respectively.

B. *Evaluation datasets*

The challenge in this competition is to estimate the trajectories of a human from the data collected by a smartphone. Nine log files, similar to those described above, were provided to the competitors. Each one of these includes data from all the same sensors, except the POSI records. From these nine log files, two of them do not include atmospheric pressure data.

For each one of these evaluation log files, the competitors must generate an estimated trajectory made of a sequence of position records every 0.5 seconds. Each record must include the timestamp, longitude, latitude, floor, and building estimates.

III. BUILDING A RADIO MAP

Our approach to this challenge is based on a WiFi fingerprinting-based method enhanced with information extracted from accelerometers' data and atmospheric pressure data. It requires, therefore, the construction of a WiFi Radio Map.

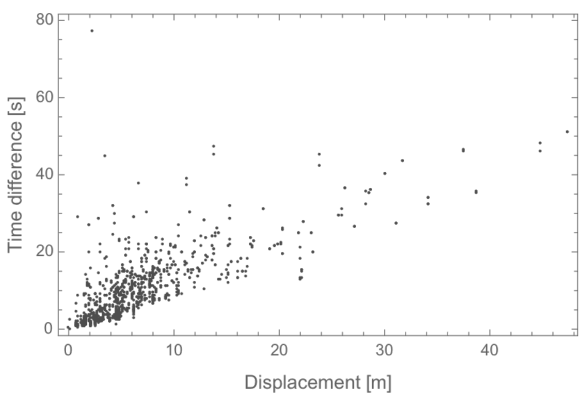

Fig. 2 Relationship between displacement and time difference in consecutive ground truth samples.

Since the WiFi records are not associated to the position (building, floor and pair of coordinates) where they were collected, this position information must be derived from other data dimensions. The obvious choice is to resort to the ground truth position data records (POSI). However, these are sparse in time and space. We, therefore, developed a method that includes three main steps: (i) the creation of additional (artificial) POSI records, inserted between pairs of the provided POSI records, to increase the spatial density of ground truth information, specially where the direct line connecting two consecutive POSI records crosses walls or other obstacles; (ii) for each trajectory, the segmentation of the data into periods of movement and periods of immobility, by processing data collected from the accelerometers; (iii) the association of a position to each one of the WiFi fingerprints (records) by processing the POSI records together with the movement profile (obtained in the second step). Each one of these steps is described in the following sections.

A. *Additional POSI records*

POSI data records are the ground truth data since these are the only records representing the absolute location of user in specific moment (time). Each POSI record is described by a timestamp, the geographic coordinates (latitude and longitude), and the building and floor ID numbers.

To collect the datasets, the organizers moved inside the 4 buildings used for the competition, creating POSI records from time to time. By plotting the POSI records over the buildings' blueprints it is possible to graphically see the users' locations sequence. Additionally, for some of the routes, the organization provided also some documents with the POSI already plotted over the blueprints and some videos that show the data collection being executed.

The sequence of POSI does not allow knowing the path followed by the organizers to collect the training dataset in detail since the POSI records are sparse in time. On some cases, a direct path (straight line) between two consecutive POSIs crosses walls or floors and, therefore, does not represent the actual path. Additionally, in some cases the time elapsed between two consecutive POSI records is large enough to allow the user to go to different places (rooms and corridors) before reaching the ensuing one.

To solve the problem of crossing walls and floors, we decided to create some additional POSI records. In several cases we inserted one or more additional POSI records in order to create a smoother (and more realistic) path between two consecutive POSIs.

The new sequence of POSIs represents the real movement of the user in the case of the routes where the video of the data collection process was available. For the routes without video information, we tried to guess the path took by the users during the data collection process.

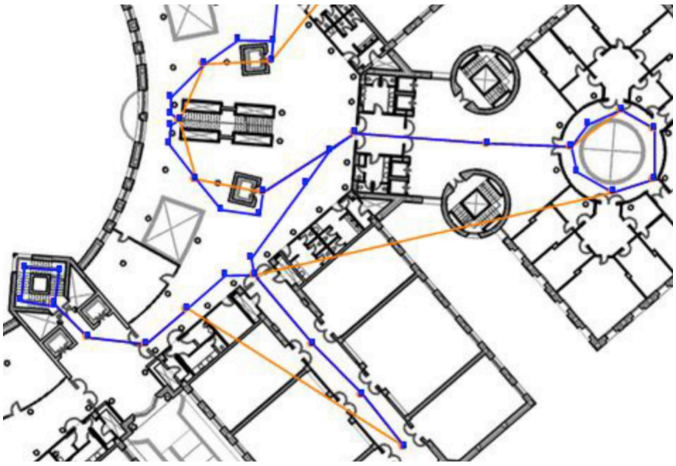

Fig. 3 Original path (orange) and corrected path after adding some additional POSI records (blue).

An example of the sequence of POSIs before and after this process is shown in Fig. 3.

B. Movement detection

To improve the construction of the WiFi radio map, our team decided to obtain information about the user's movement. In particular we want to determine the time instants in which the user is moving or is stopped. All the WiFi fingerprints collected during the time when the user is stopped must be associated with the same position (section III-C). Additionally, we intend also to use this information to improve the position estimation procedure (section IV-E).

The movement information was obtained using the data records related to the accelerometer sensor (ACCE). We begin to calculate the acceleration magnitude using the accelerations in the X, Y and Z axes included in ACCE data records. Then and after removing the 'gravity' (DC component) of the acceleration magnitude signal, we applied a one-dimensional median filter to the resulting signal in order to supress the noise, and a Butterworth filter to eliminate impossible frequencies [8]. Finally we counted the steps by finding the peaks in the resulting signal.

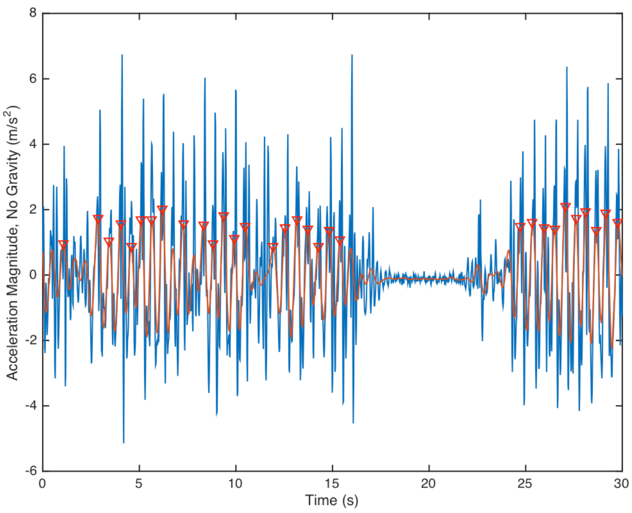

Fig. 4 Step detection example.

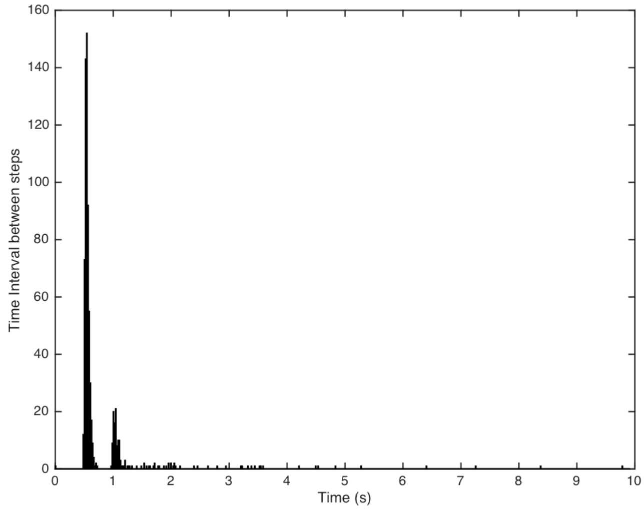

Fig. 5 Histogram built from UJITI_R2_NEXUS5 route.

In Fig. 4, an example of step detection process results is presented. The blue line is the acceleration magnitude signal after the DC component has been removed. The red line is the signal after the filters have been applied. And finally the output of the steps detector is represented by small red triangles.

Using the step detector output, we build a histogram with the time interval between steps. The idea is to determine the most common interval between steps of the user carrying the smartphone. An example of these histograms is presented in Fig. 5. We then use the most frequent time interval between steps to determine if the user is moving or stopped. If the time interval between two steps is greater than the most frequent time interval between steps (plus a safety margin) we consider the user as being stopped during this time interval. Otherwise we consider the user is moving and increment a steps counter. The output of the whole process is a movement profile indicating when the user was stopped or moving. The number of steps detected during each moving segment is also provided. Table 1 shows the number of steps and the most frequent time interval between steps calculated for each one of the evaluation datasets.

TABLE I. EVALUATION DATASETS STATISTICS OBTAINED FROM MOVEMENT DETECTION PROCEDURE

	Eval 01	Eval 02	Eval 03	Eval 04	Eval 05	Eval 06	Eval 07	Eval 08	Eval 09
#Steps	118	470	1191	517	706	795	1201	796	795
Steps Time Interval (seconds)	0.5	0.56	0.64	0.5	0.6	0.64	0.64	0.61	0.64

C. Building the WiFi radio map

The radio map constructed by our team consists of a single WiFi fingerprint map for all the 17 datasets provided. All WiFi fingerprints must be associated with a geographic position. The coordinates of each position are obtained from the POSI records provided in the log files and also from the new additional POSI records created manually, as described above. A WiFi fingerprint is a vector of receiver signal strength (RSS) for all known wireless access points (AP1, ..., APn). Fingerprint information is generated from the WIFI records provided in the log files. The process of radio map construction is therefore structured in two distinct phases. In the first phase the WiFi fingerprints are constructed. In the second phase all the gathered fingerprints must be associated with a position. As shown before, this is a more complex task, since there is no direct temporal correspondence between the provided POSI and WIFI records. Relations must be inferred based on time information.

A total of 2971 fingerprints were built from the log files in the first phase. Files were parsed and WIFI records extracted one by one. Each MAC address present in the record is first normalized to a canonical format and then indexed in a map, associating it with a unique sequential number. In the end a total of 816 unique AP addresses, numbered from 1 to 816, were identified. This is the dimension of the each fingerprint vector. In the log files, each WIFI record contains information

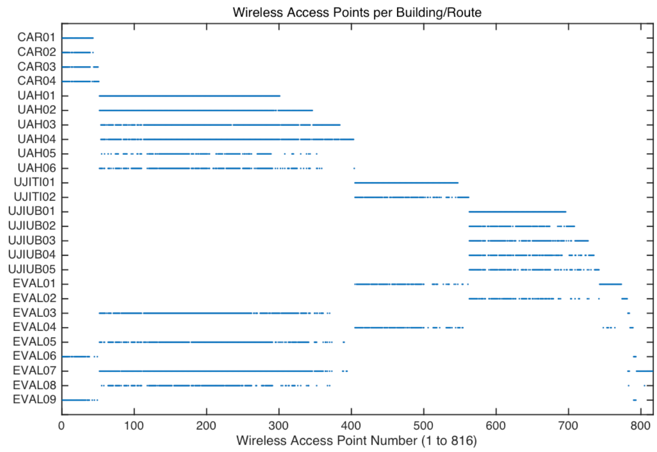

Fig. 6 AP presence per building and route.

of a single AP, and when more than one AP was observed in the same position, multiple lines were generated with the same timestamp. We can therefore aggregate all the measurements with the same timestamp, in the same fingerprint vector. At this point, we can already plot a graph showing the relationship between the identified APs and the 17 datasets and 9 evaluation sets, as shown in Fig. 6. Evaluation datasets are easily mapped to the corresponding buildings. We can also notice that there are several new MAC addresses identified in the evaluation sets that were not present in the 17 datasets.

At the end of the first phase, the information in the fingerprint map consists of geographic positions with no fingerprint directly associated with them, and fingerprints with no position associated with, as shown in Fig. 7.

Four strategies were identified to associate them: i) associate to each known geographic position its nearest (in time) fingerprint; ii) associate to each fingerprint the coordinates of the nearest known position; iii) do a linear interpolation, between known positions, based on time information; iv) do the linear interpolation, but combine that with the movement information that represents if the user is moving or stopped at a given location.

Long.	Lat.	Floor	Build.	Time	AP 001	AP 209	AP 210
?	?	?	?	2.015	100	-87	-86
?	?	?	?	6.058	100	-86	-85
-3.3484	40.5127	0	20	8.484	?	?	?
?	?	?	?	10.088	100	-85	-86
?	?	?	?	13.987	100	-87	-87
?	?	?	?	17.865	100	-84	-83
?	?	?	?	21.749	100	-82	-82
?	?	?	?	25.777	100	-86	-85
-3.3483	40.5129	0	20	29.264	?	?	?
…	…	…	…	…	…	…	…

1) nearest fingerprint 2) nearest POSI 3) linear interpolation

Fig. 7 Strategies to associate fingerprints and positions.

The first solution is simple and reliable, but results in a reduced map of only 1669 fingerprints, one for each POSI provided or added. The second solution results in a bigger map with a total of 2971 records, one for each fingerprint. But errors may be introduced, depending on the amount of time between each WiFi record and the nearest POSI. All fingerprints are associated with one position that is the nearest one in time, but the nearest one in time may be really distant in space, introducing more imprecision in the radio map. The third solution tries to minimize this effect by doing a linear interpolation of the coordinates in space according to the time. The fourth solution improves it further with the movement information extracted from the accelerometers' data. If we know that the user is immobile at a given position, we can associate all fingerprints collected there, while he was stopped, with that position. We decided to adopt this fourth solution to construct the final version of our WiFi radio map.

Fig. 8 shows the positions and fingerprints obtained for building UAH in one of the routes provided (UAH_R2_S4). The figure highlights the differences between using or not using interpolation. The green circles represent all the known positions (POSI records) for this route, while the red triangles

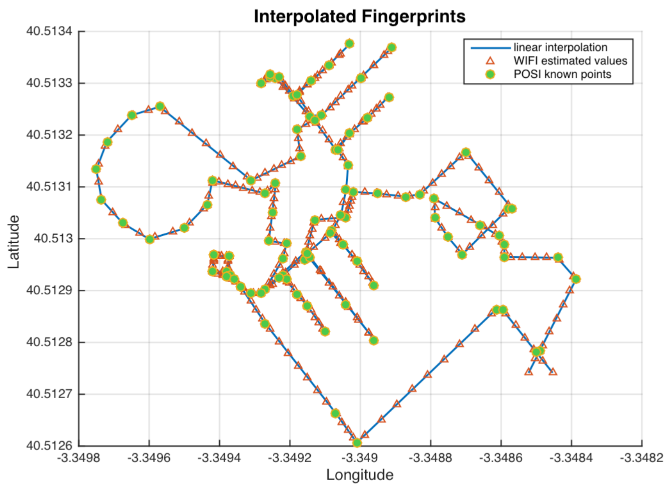

Fig. 8 Estimated fingerprint positions using linear interpolation.

show the interpolated positions for the fingerprints.

IV. Position estimation

The challenge associated to this competition is to estimate the trajectory of a human by processing the data collected from the several sensors available in smartphones. No matter the time elapsed between samples (provided), the competitors must represent the estimated trajectory by a sequence of position records generated every 0.5 seconds.

Our approach for estimating the trajectory is based on WiFi fingerprinting. We, therefore, started by creating a radio map as described in section III. To estimate the trajectory we start by estimating the position associated to each and every one of the WiFi records in the evaluation log files: first we estimate the building (section IV-B), then we estimate the floor (section IV-C), and finally we estimate the position (pair of coordinates) within the floor (section IV-D). For estimating the floor, one of two methods is used, depending on the availability of data from the atmospheric pressure sensors. For estimating the coordinates, a simple KNN approach is used.

Finally, from the sequence of positions estimated for each WiFi record, we generate the final estimated trajectory by using linear interpolation. Three methods are used, trying to explore the information provided by movement profile obtained from the accelerometers' data.

Since this is an "off-site" competition, data about each one of the trajectories is available at once, and can be processed as a whole without respecting causality. Processing data without respecting causality (i.e. without emulating real-time processing) limits the use of the position estimation methods to non-real-time applications. In the methods described in the next sections, some of them do not respect causality, being clearly identified.

A. Names and conventions

Throughout this paper, the following names and conventions are used:

R – The radio map used for positioning estimation

fp_i – Denotes fingerprint i of a radio map

fp_0 – Denotes a evaluation fingerprint (unknown position)

AP_i^n – Denotes the n^{th} strongest AP in fingerprint fp_i

$rssi_i^j$ – Denotes the RSSI value of the i^{th} AP in fp_j

k – The number of neighbours in k-nearest neighbours approaches

p – The estimated position (pair of coordinates)

f – The estimated floor

b – The estimated building

ps_i – Denotes the i^{th} atmospheric pressure sample

B. Building estimation

For estimating the building associated to each fingerprint, the method described in [7] has been adopted.

Given a fingerprint fp_0, the corresponding building (b) is estimated as follows:

1. Take AP_0^1, the strongest AP observed in fp_0.
2. Build R', a subset of the radio map R, with all the samples where the strongest AP is AP_0^1 (filtering).
3. If R' is an empty set, repeat steps 1 and 2 for the 2nd, 3rd, ..., strongest AP in fp_0.
4. Count the number of samples in R' associated to each building and set b to the most frequent building (majority rule).

This procedure is very efficient from the computational effort point of view since no similarities between fingerprints need to be computed. It also proved to be very reliable, with a precision of 100% in estimating the correct building [7]. Similar results were obtained with the data provided for this competition.

C. Floor estimation

Since the provided data includes samples about atmospheric pressure, two methods to estimate the floor were considered, one based on the pressure data, and one based solely on the WiFi data. This last one has been introduced in [7], and is described next.

Given a fingerprint fp_0 and the corresponding estimated building (b), the corresponding floor (f) is estimated as follows:

1. Build R', a subset of R, with all the samples where the building is b (filtering).
2. Build R'', a subset of R', with all the samples where the strongest AP is AP_0^1, AP_0^2 or AP_0^3 (filtering).
3. If $\#(R'') < n$, then $R'' = R'$, where #(.) denotes the cardinality of a set, and n is a parameter.
4. Compute the similarity, $S()$, between fp_0 and all the fingerprints in R''.
5. Take the $k1$ samples in R'' that are the most similar to fp_0.
6. Count the number of samples, from within the $k1$, associated to each floor, and set f to the most frequent floor (majority rule).

In step 4. above, the similarity function $S()$ is the Manhattan distance defined as:

$$S(\mathrm{fp}_1, \mathrm{fp}_2) = \frac{1}{N} \times \sum_{i=1}^{N} |\mathrm{rssi}_i^1 - \mathrm{rssi}_i^2| - 2 \times C \tag{1}$$

where N is the total number of APs observed in fp_1 and/or fp_2, and C is the number of APs that were observed in both fp_1 and fp_2 (common APs). For missing APs, in fp_1 or fp_2, a default RSSI value was used.

The process described above estimates the floor given an single and independent fingerprint fp_0. However, the sequences of fingerprints were collected along a single trajectory and are, therefore, not independent since a pedestrian cannot move between floors within a very short time period. This fact enables the detection of erroneous floor estimates (outliers) in some cases, such as when the sequence of floor estimates is floor1, floor3, floor1 and the time elapsed between the first and third estimates is very short (a few seconds). We explored this fact to develop an outlier detection method and to try to correct those erroneous floor estimates. The method is very simple: firstly, potential outliers are detected, and then the estimated floor is corrected to the floor estimate associate to the previous fingerprint if the probability of being the correct floor is lower than a given threshold (*P2*), or corrected to the second most probable floor if that same probability is lower than another threshold (*P1*), with *P1* > *P2*; otherwise, the outlier is not corrected.

ALGORITHM 1 DETECTING AND FIXING OUTLIERS IN FLOOR ESTIMATES.

```
if fi ≠ fi-1 AND fi ≠ fi+1 AND Δt < T then {
    a potential outlier was found
    if Pi < P1 then
        if Pi < P2 then
            fi ← fi-1
        else
            fi ← fi^2
}
```

This process is described in Algorithm 1, where Δt is the time elapsed between f_{i-1} and f_{i+1}, *T* is a parameter (we used 15 seconds), P_i is the probability associated to the i^{th} floor estimate (obtained from the majority rule described above), *P1* and *P2* are parameters (we used 0.85 and 0.3, respectively), and f_i^2 is the second most probable floor (obtained also from the majority rule). Fig. 9 shows an example of the results obtained in estimating the floor profile for log file 07.

In the example of Fig. 9, there are two cases (before t=50 s) where the floor estimation process returned floor 1 as the most probable one (in orange); however, most of the estimates around these two point to floor 0; after the outlier detection and correction procedure, these two estimates were corrected to floor 0 (in green). However, there are other cases (e.g. around t=200 s) where the apparent outliers were not detected. Please note that it is not possible to calculate the precision of this process since the provided data do not include enough ground truth about the floor profile.

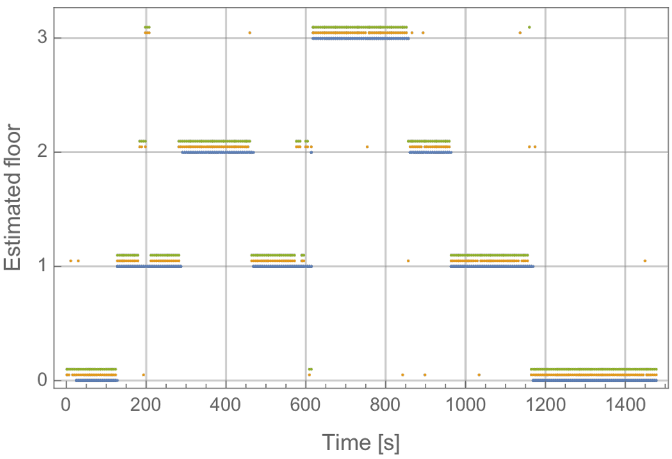

Fig. 9 Floor estimation: from pressure data (blue); from WiFi data (orange); from WiFi after outlier detection and correction (green).

Alternatively, a floor estimation method based on the atmospheric pressure data was developed. This method processes all the available pressure data records at once and also takes as input the number of floors for each building.

Given the sequence of pressure samples (*ps*), the floor for each given time *t* is estimated by first taking the derivative of the pressure signal (*dps*):

$$\mathrm{dps}_i = \frac{1}{w1} \times \left(\sum_{j=0}^{w1-1} ps_{i-j} - \sum_{j=0}^{w1-1} ps_{i-j-d} \right) \qquad (2)$$

where *w1* is a parameter controlling the number of consecutive samples used to represent the pressure (mean of the *w1* latest samples) and *d* is the delay. The differentiated signal is then smoothed by a low pass filter (moving average), defined as:

$$\mathrm{sps}_i = \frac{1}{w2} \times \sum_{j=0}^{w2-1} ps_{i-j} \qquad (3)$$

where *w2* is a parameter controlling the amount of smoothing. The smoothed signal is then compared against a set of thresholds to detect floor changes. Whenever the smoothed differentiated pressure crosses a negative threshold, the user is assumed to have moved to an upper floor; whenever a positive threshold is crossed, the user is assumed to have moved to a lower floor. The number of floors per building (input to the process) is used to shift the estimated floor profile up or down to avoid having a number of floors higher than the actual number of floors in the building. The complete process is illustrated in Fig. 10.

In Fig. 9 there is a comparison between the two methods, the one based on the pressure data (in blue) and the one based on the WiFi data (in green). Most of the time, the solution based on the pressure data seems more precise than the other one. Similar results were obtained for the other 8 log files.

D. Coordinates estimation

The procedure used to estimate the coordinates associated to a given fingerprint fp_0 is a simple KNN solution, similar to the one presented in [7], and is as follows:

1. Build *R''*, a subset of *R*, with all the samples where the building is *b* and the floor is *f* (the building and floor estimated in the previous steps) (filtering).
2. Compute the similarity, *S()*, between fp_0 and all the fingerprints in *R''*.
3. Take the *k2* samples in *R''* that are the most similar to fp_0.
4. Compute the estimated coordinates as the centroid of the *k2* samples.

Here, the similarity function used in step 2. is also the one defined in equation (1).

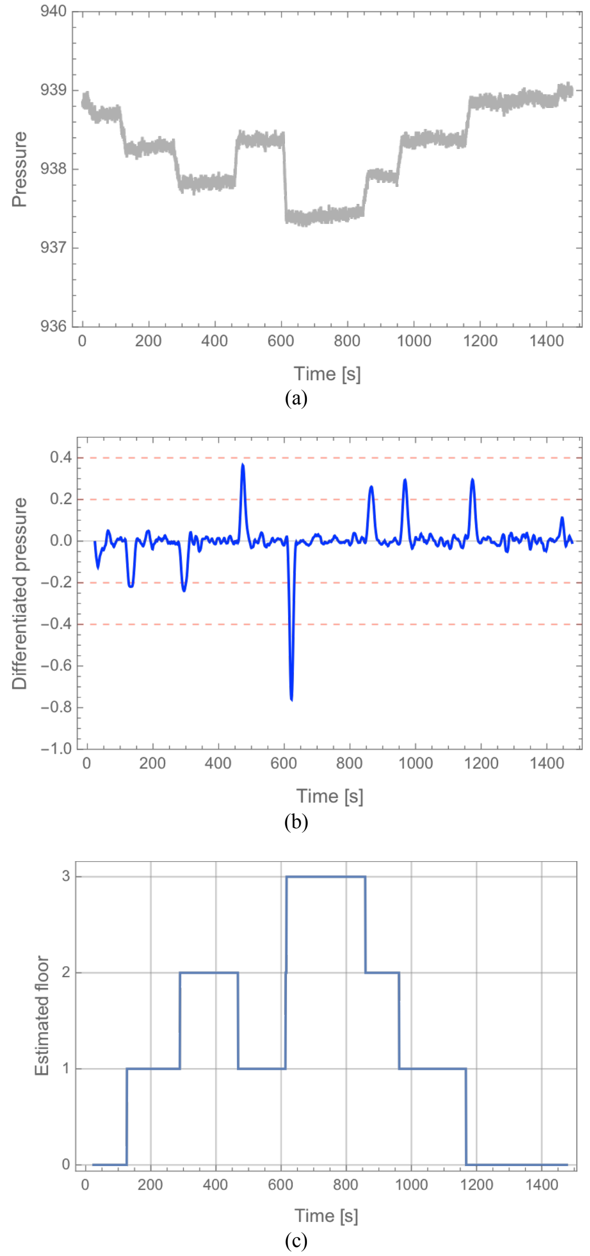

Fig. 10 Floor estimation process based on the atmospheric pressure data: (a) pressure profile; (b) differentiated and smoothed pressure profile (with thresholds); (c) final estimated floor profile.

E. Final trajectory

As defined in the competition rules, the final trajectory is a sequence of position estimates with a sampling rate of 0.5 seconds. The methods described in the previous sections can be used to estimate the position (building, floor and coordinates) associated to each one of the available WiFi fingerprints. However, these fingerprints were collected with an average sampling period of 4.7 seconds. Therefore, a method is required to estimate the positions along the trajectory, every 0.5 seconds, from the much more sparse (in time) WiFi fingerprints. Three methods have been used for this purpose.

<u>Trajectory estimation - variant 1:</u>

A simple solution to generate the trajectory is to resort to linear interpolation, where the estimated position $p_j=(x_j,y_j)$ for a given time instant t_j is obtained by the linear interpolation of the positions associated to the two nearest WiFi fingerprints (one before (W_{i-1}) and another after (W_i) t_j), weighted by the timestamps associated to the WiFi fingerprints, given by (see Fig 11a):

$$x_j = \frac{t_i - t_j}{t_i - t_{i-1}} \times x_{i-1} + \frac{t_j - t_{i-1}}{t_i - t_{i-1}} \times x_i \tag{4a}$$

$$y_j = \frac{t_i - t_j}{t_i - t_{i-1}} \times y_{i-1} + \frac{t_j - t_{i-1}}{t_i - t_{i-1}} \times y_i \tag{4b}$$

When pressure data is available, the estimated floor is obtained directly from the estimated floor profile (nearest sample). Otherwise, estimated floor is the one associated to the previous WiFi fingerprint.

<u>Trajectory estimation - variant 2:</u>

Since accelerometers' data is available and we can derive a mobility profile (see section III-B) along the trajectory, there is a potential to improve the position estimates along the trajectory whenever the user is not moving. One possible approach is to combine the position estimates associated to all the WiFi fingerprints collected while the user was immobile into an average position (see Fig. 11b), and assign that position to all the points of the trajectory inside the immobility window.

<u>Trajectory estimation - variant 3:</u>

One alternative to explore the mobility profiles is to estimate the position associated to each WiFi fingerprint within the immobility periods using a merged fingerprint instead of

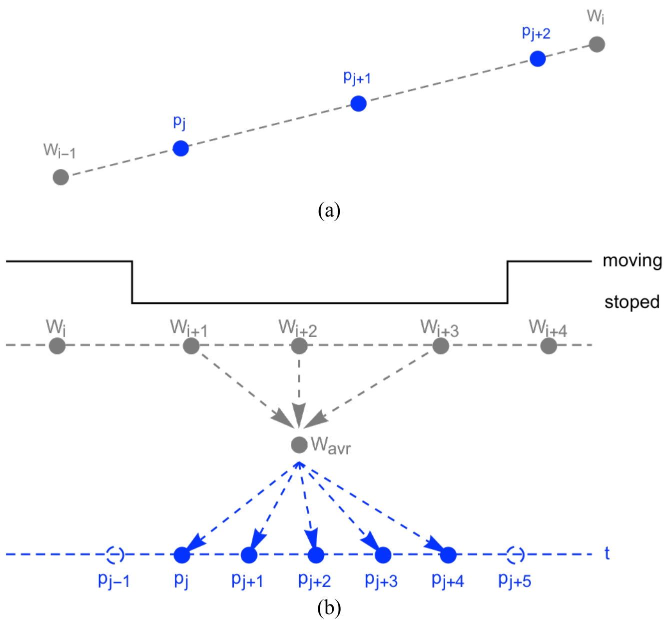

Fig. 11 Generating the final trajectory: (a) using linear interpolation; (b) exploring the immobility periods.

using each fingerprint independently. In this approach, the position associated to W_{i+1}, W_{i+2} and W_{i+3} (see Fig. 11a) are estimated using the method described in section IV-D with fp_0 being the result of merging fp_{i+1}, fp_{i+2} and fp_{i+3}. The trajectory is then estimated as in the first variant.

V. RESULTS

The processes described in the previous sections were used to build the radio map from the provided datasets with POSI data, and to generate the estimated trajectories for each one of the log files without POSI data. Since this paper was written before the actual competition, i.e., before our results are evaluated by the competition organizers, no information is yet available about the exact performance of the proposed solutions, and no metrics can be used to compare the three variants described in the previous section. While developing our solution and adjusting the several parameters, we resorted to visual analysis of the estimated trajectories to assess the quality of the results. One of those visualizations is represented in Figure 13, where the estimated trajectory obtained for log file 04 is plotted (one colour per floor). Figure 12 represents the corresponding results of the floor estimation process (no pressure information is available to this log file), where the outlier detection and correction method fixed a few estimates.

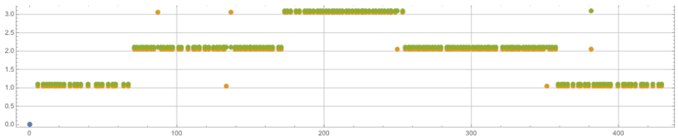

Fig. 12 Obtained results for log file 04: floor detection.

VI. CONCLUSIONS AND FUTURE WORK

This paper describes the approach adopted by the UMINHO team to address the challenges of the IPIN 2016 indoor position competition. From within the provided data, our solution makes use of four types of records: WIFI, for building a fingerprinting radio map; pressure (PRES) for floor detection; accelerometer (ACCE) for movement detection, used in building the radio map and in generating the estimated trajectory; and position ground truth (POSI) for building the radio map. A method is proposed to build the radio map, with four variants. One of them was used to generate the final radio map. A few methods are described for building detection, floor detection (two variants) and position (coordinates) estimation. Finally, three solutions are discussed for estimating the final trajectories from the evaluation files.

The results obtained with this work have been evaluated through visual inspection and look promising. Final evaluation results, using clear and objective metrics, will be possible after the competition organizers release the ground truth or the evaluation results. Based on the evaluation results, some of the used methods will be reviewed and fined tuned for better performance.

ACKNOWLEDGEMENTS

This work has been supported by COMPETE: POCI-01-0145-FEDER-007043 and FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2013.

REFERENCES

[1] http://evaal.aaloa.org (visited July 2016)

[2] http://www.ucalgary.ca/ipin2015/call-papers/call-competition (visited Sept 2016)

[3] http://evaal.aaloa.org/2015/competition-home (visited Sept 2016)

[4] P . Bahl and V . N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in Proc. IEEE INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. , Tel Aviv, Israel, pp. 775-784, 2000.

[5] http://evaal.aaloa.org/2015/call-for-competitions (visited Sept 2016)

[6] http://www3.uah.es/ipin2016/cfc.php (visited Sept 2016)

[7] Adriano Moreira, Maria João Nicolau, Filipe Meneses, and António Costa, "WiFi Fingerprinting in the Real World - RTLS@UM at the EvAAL Competition", Proceedings of the Sixth International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2015.

[8] Bujari, A.; Licar, B. & Palazzi, C. E. (2012), Movement pattern recognition through smartphone's accelerometer., in 'CCNC' , IEEE, , pp. 502-506

Fig. 13 Obtained results for log file 04: estimated trajectory.