

# A Model-Based WiFi Localization Method

Dik Lun Lee and Qiuxia Chen  
 The Hong Kong University of Science and Technology  
 Department of Computer Science and Engineering  
 Clear Water Bay, Hong Kong  
 {dlee, chen}@cse.ust.hk

**Abstract**—Due to the proliferation of WiFi access points, indoor localization methods based on WiFi signal strengths are becoming more and more attractive because they don't require additional infrastructural costs beyond the existing WiFi infrastructure. Many research projects were proposed based on this approach, but most of them only focused on the processing of the signal strength data to obtain the user locations. Very little attention was paid to the mobility patterns of the users. In this paper, we propose a method based on (i) knowing the floor model, (ii) continual tracking of user locations and (iii) back-tracking from the current location to previous locations to resolve localization ambiguities. We implemented the system in a life environment and performed experiments to measure the localization accuracies. We found that our method identified all test paths accurately with the exception of a challenging case where two locations were connected together with a thin wall. We discuss ways to handle this situation.

**Index Terms** — location identification, location tracking, mobility, floor model.

## I. INTRODUCTION

Due to the decreasing cost and the ease of installation of access points, indoor localization using WiFi signal strengths are becoming more and more popular. Many research projects had been proposed based on this approach, but most of them were based on a single set of signal strengths obtained at the time localization was performed. As such, they focused on signal processing methods and triangulation techniques. Very few approaches made use of continuous monitoring of user locations to increase localization accuracy [5]. In this paper, we propose a backtracking technique to improve the accuracy of the localization results. We conducted experiments in a life setting to show the accuracy of the system and discussed challenges we faced in the experiment.

In the rest of this paper, we will give a brief overview of recent WiFi-based localization methods in Section 2. In Section 3, we introduce the basic idea of our approach called WHAM! (Where Am I). In Section 4, we present the system architecture. In Section 5, we evaluate the accuracy of our system.

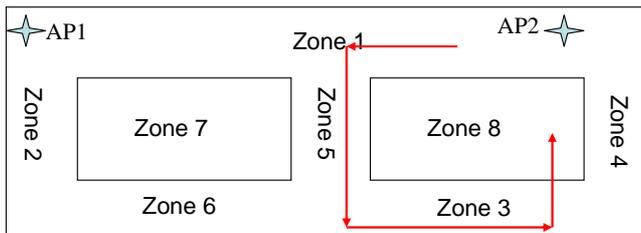
## II. PREVIOUS WORK

### *RADAR: An In-Building RF based User Location and Tracking System*

Developed by Microsoft Research RADAR makes use of WiFi signal strengths for indoor localization [1]. It is based on two principles about WiFi signal strengths; the WiFi signal strength decreases when the transmitter and the receiver are further apart and when there are obstacles between the transmitter and the receiver. Base on these two principles, WiFi signal strength readings were measured on various points of the test site. The readings were recorded to a database. When a location query was submitted by the user, the system compared the user's current signal strength values with the signal strength values in the database. Then the system will deduce a most probable location and return it to the users.

Several projects were conducted based on matching of WiFi signal strength values [7]. In general, it was found that WiFi signal strengths are unstable in that the signal strengths at the same location changes with time, temperature and objects around it. Thus, it is hard to achieve good accuracy based on pure matching of signal strength. Ho, et. al., proposed a method to retrain the system when the user finds that the estimation location is wrong [3].

Most existing methods only make use of the signal strengths measured at the user's location for estimating what the user's actual location is. Few, if any, made use of the user path and movement to enhance localization accuracy [5]. This paper presents a new approach which continuously records the signal strengths received at the user client, and disambiguates the user's location by



**Figure 1** A floor plan with two rooms.

backtracking to the user's previous locations in the floor model and eliminating candidate locations that are not likely to be reachable from the previous locations. Note that we assume that the user can enable or disable the monitoring service as he/she enters the monitored area; privacy concerns are beyond the scope of this paper.

### III. WHAM!

This paper proposes a method called WHAM! (Where Am I), which is part of an indoor navigation and semantic location modeling project. In our approach, the localization system continuously monitors the user's positions in an area. When a localization request is received from the user, the system will estimate the user's actual location using a standard WiFi localization method. If a unique location is obtained, the system will return the estimated location to the user. However, when ambiguity arises, meaning that the system finds more than one location satisfying the measured signal strengths, it will use the user's previous locations to disambiguate the set of candidate locations.

Since continuous computation of the user's location is very time consuming, WHAM! will record only the signal strength values of the user at regular interval. This is not a very time consuming operation. When previous locations are needed for disambiguation, WHAM! will "backtrack" and compute from the stored signal strengths the immediate preceding location of the user. By estimating the likelihood for the user to move from the previous locations to the candidate locations, WHAM! can discard candidate locations that are not likely to be reached from the user's previous locations. If a conclusive answer still cannot be obtained, the system can back track one more step until a clear answer is obtained or a system-defined backtracking level is exceeded.

To facilitate disambiguation, our method requires a semantic location model to represent the possible locations and paths within the indoor environment [4]. For the purpose of this paper, a floor model of the indoor environment is sufficient. We assume that it is possible to

**Table 1**  
SIGNAL STRENGTH VALUES OF DIFFERENT ZONES

Zone	Single Strengths of AP1	Signal Strengths of AP2
1	4	4
2	4	2
3	2	3
4	2	4
5	3	3
6	3	3
7	1	1
8	1	1

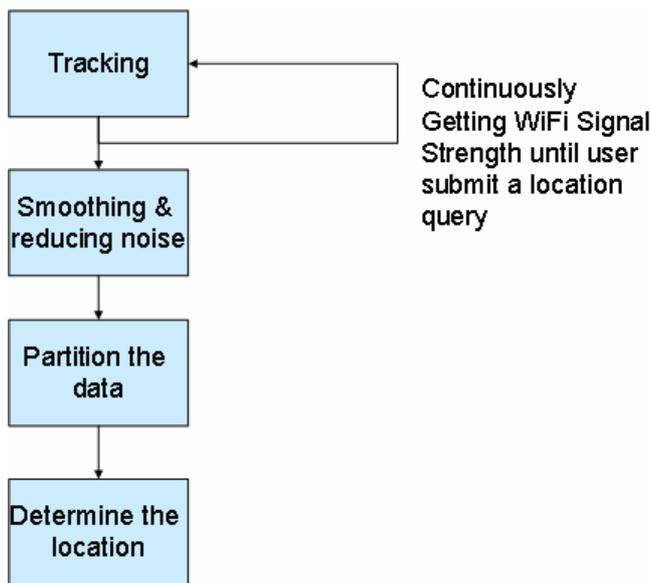
query the zones (e.g., rooms and corridors) and the paths between the zones within the floor model as is possible in a semantic location model [4].

**Figure 1** illustrates how WHAM! works. The floor plan consists of eight zones, representing two rooms and the corridors around them. There are two access points, AP1 and AP2, at the upper left and upper right corners, respectively. Notice that a corridor can be represented by more than one zone (e.g., Zone 3 and Zone 6) according to the need of the application. The signal strengths from AP1 and AP2 are measured for each zone, which is shown in Table 1. Notice that the signal strength values are for illustrative purpose only and that WHAM! is not very sensitive to the actual signal strength values but instead their relative values. For example, since there is no obstacle between the two APs and any point within Zone 1, the signal strengths measured in Zone 1 are constantly at a rather high level. On the other hand, Zone 2 will receive much higher signal strength from AP1 and a weak signal from AP2 because of the blockage of Zone 7. It is clear that because of the knowledge of the floor model, WHAM! does not rely very much on the absolute signal strengths for computing user locations.

A user traversed the path Zone 1  $\rightarrow$  Zone 5  $\rightarrow$  Zone 3  $\rightarrow$  Zone 8 and submitted a location query in Zone 8. Suppose the recorded signal strengths when the user's first three locations are as shown in Table 2. If only the final signal strength data were considered, it is very difficult to determine whether the user was in Zone 7 or

**Table 2**  
SIGNAL STRENGTH VALUES OF A USER AT DIFFERENT TIMES

Time	Single Strength of AP1	Signal Strength of AP2
t=1 (Zone1)	4	4
t=2 (Zone5)	3	3
t=3 (Zone3)	2	3
t=4 (Zone 8)	1	1



**Figure 2** The structure of WHAM!.

Zone 8. However, if we backtrack to  $t=3$ , the signal strength values of AP1=2 and AP2=3 uniquely identify the user's location to be in Zone 3. Thus, further backtracking is not needed. With the knowledge that the user was in Zone 3, we can infer that the user's current location is Zone 8, not Zone 7, because it is impossible for the user to reach Zone 7 from Zone 3.

#### IV. SYSTEM STRUCTURE

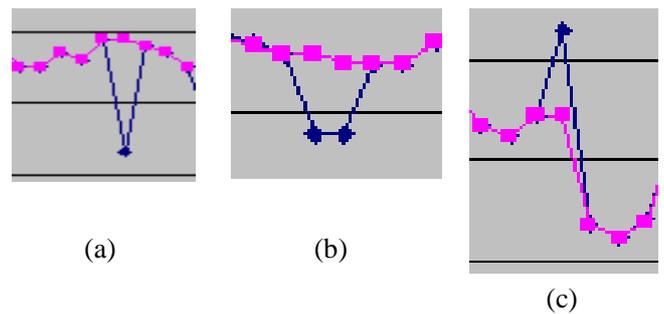
WHAM! was implemented in J2SE. It mainly includes four modules, namely, Tracking, Smoothing, Partitioning and Determining the zone. Figure 2 shows the structure of our system. Details of the four modules are described below.

##### *Tracking*

WHAM! continuously records signal strength values from each access point at the user's location with fixed time interval when the user is walking around the floor. When the user submits a location query, WHAM! will process the collected signal strengths to report his/her current location.

##### *Smoothing*

The variation of the collected signal strengths is large. In order to minimize the effect of noise and outliers, we need to smooth our collected data before processing so that it can increase the accuracy of finding the cut-off points to partition the data (see next step). After analyzing the raw data we collected, we observed a few patterns that could be smoothed. In Figure 3(a), there is a drop in the

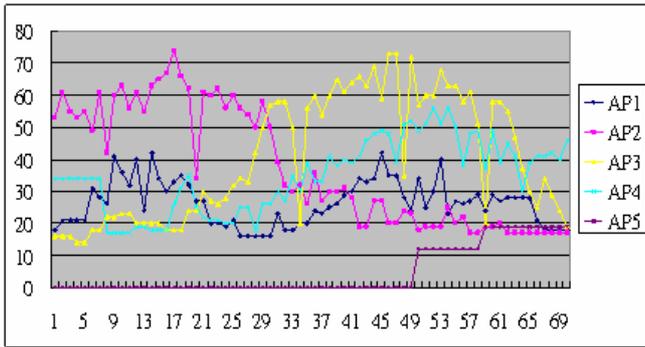


**Figure 3** Outliers that can be smoothed. The blue line represents the raw data. The pink line represents the data after smoothing.

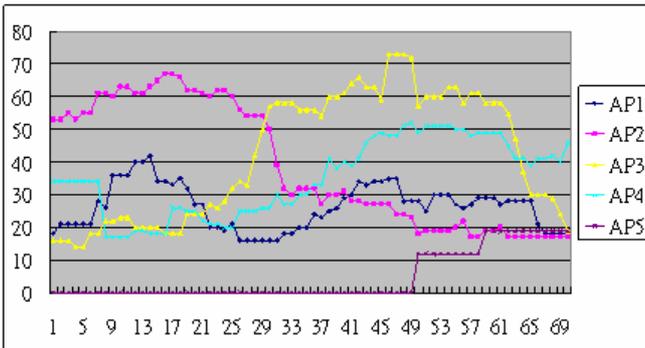
data sequence. The drop can be smoothed out because if the difference between the signal strengths on both sides of the drop is small, it is more likely that the data are in the same zone. The same reason applies to Figure 3(b), where there are two consecutive drops. For three or more successive drops, it is difficult to say whether they are outliers or they are in a different zone. Figure 3(c) shows an outlier at the cut-off point. This point can be treated as an outlier because it is not possible to have a sudden increase followed by a large drop. Figure 4 and Figure 5 show the difference between the data before and after smoothing.

##### *Partition*

WHAM! collects the signal strengths periodically when the user is walking around an area, thus generating a long sequence of signal strength data. We need to partition the long sequence of data before we can map the partitions into zones on the floor plan. The purpose of partitioning is to separate the collected data to several segments such that the data in each segment is more likely to belong to the same zone. We identify the segments by finding the cut-off points. After partitioning is done, we can analyze the data in each segment to determine which zone it belongs to. It is more reliable and accurate to analyze the data within a segment together than using individual data points.

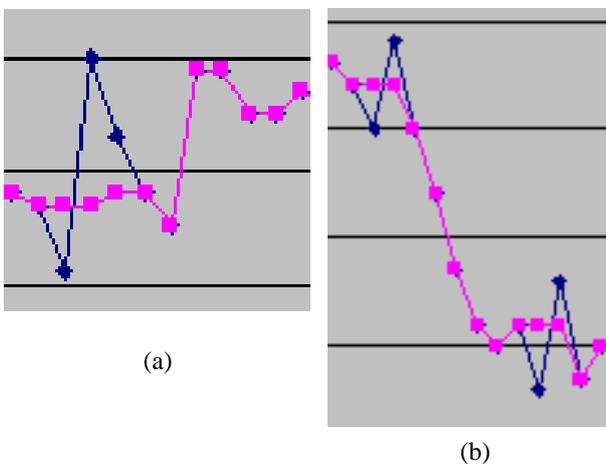


**Figure 4** The measured signal strengths before smoothing (signal strengths against time).

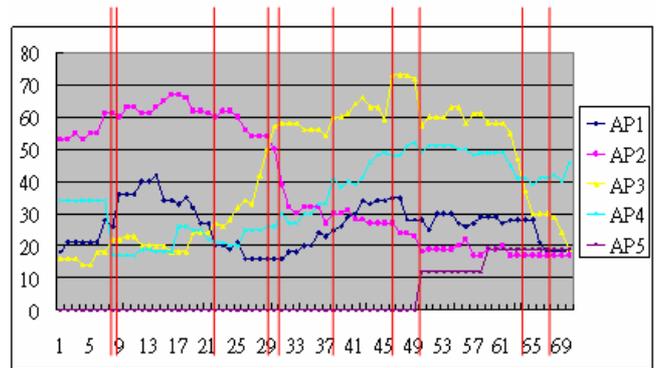


**Figure 5** Signal strengths after smoothing (signal strengths against time).

Figure 6 shows the cases for two zones with a cut-off region. In Figure 6(a), there is a sharp change in the smoothed signal data (pink curve), indicating that the user is likely moving from one zone to another. In Figure 6(b), however, it is hard to find the cut-off only by comparing two successive points because the points gradually decrease in the cut-off region. To resolve this problem, we use cumulative mean to find such cut-off points. Cumulative mean is the mean value of the data points inside a window. Cumulative means of consecutive



**Figure 6** Cut-off regions.



**Figure 7** Partitions of the collected signal strength values (signal strengths against time).

windows are compared to find the cut-off point. We assume that the number of data points within the same zone is large enough that the data points in the cut-off region do not vary the mean value too much.

In our system, we first find the cut-off points based on the signal strength data sequence of each access point. Then, we union the cut-off points for all access points to partition the data. Figure 7 shows the partitions of one of the collected data sequence. There may be some small partitions because the cut-off points in different access points may not fall at the same point. Finally, we merge small neighboring partitions because calculation based on small partitions is not reliable.

*Zone Determination*

The final step is to map the signal segments into zones. We measured the maximum and minimum signal strengths of each zone as an input to WHAM!, which will determine the possible zones of each segment based on these min/max values. For each segment, WHAM! calculates the mean value of the signal strengths of each access point, and then associate with each zone a score showing how many mean values of the segment fall into the min/max ranges of this zone. The ones with the highest score are the possible zones of this segment.

As described earlier, WHAM! will start from the last segment and backtrack, if necessary, along the path. By comparing the connectivity of the graph and the possible zones in each segment, WHAM! can eliminate some of the identified zones to get the correct answer. Figure 9 illustrates the idea. The graph corresponding to the floor plan will have a self loop at each node because the segments do not necessarily map one-to-one to the zones traversed. It is possible to partition the data within the same zone into two or more segments. Sometimes, it is possible to have more than one valid answer. For each

possible answer, our system will sum up the average possible scores in each segment along the path, and then rank the results by their total possible scores. Further, it is possible to have no valid answer. Then, WHAM! will report the zone with the longest backtrack path.

V. EXPERIMENT

Our experiment was carried out in the Laboratory area on 4th floor of the Department of Computer Science and Engineering. The layout of the floor is shown in Figure 8. Its size is 43.5m by 40.5m, with an area of 1761.75 sq. m. We used five access points in our experiment at the locations shown in Figure 8. We separated the lab area into nine zones as shown in Figure 8. The paths we used to test our system are shown in Table 3. The system collected one signal strength sample per second when the user is walking along the paths.

**Table 3**  
THE PATHS FOR TESTING

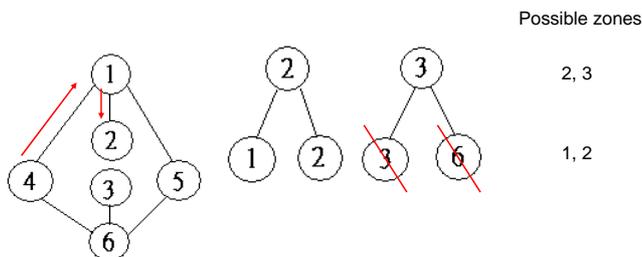
Paths	
1	4->9->5
2	4->1->2
3	5->9->7
4	4->8->3
5	5->1->4->9->7
6	4->8->6
7	4->8->5
8	5->8->3

VI. RESULT AND DISCUSSION

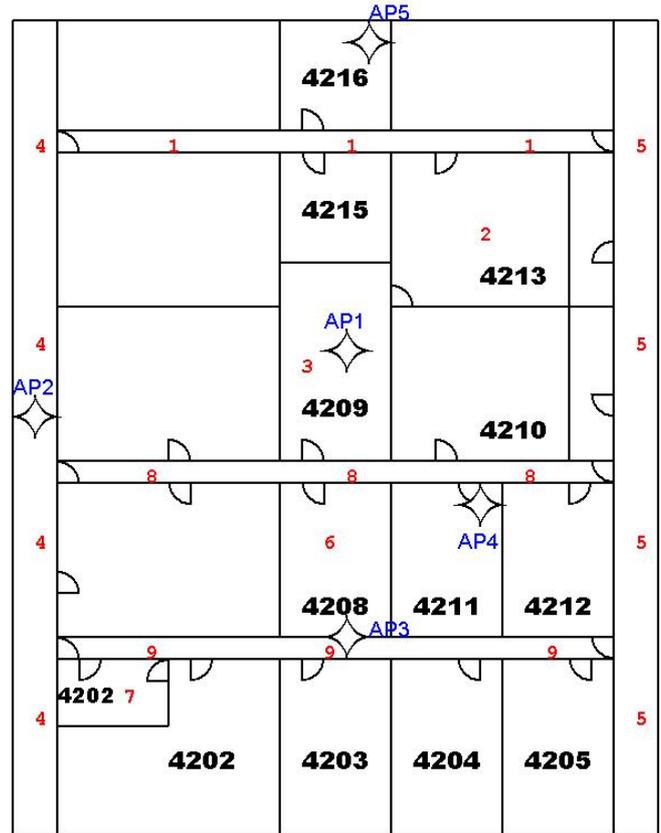
After a user submitted a location query to the system, there are 3 results that can be used to evaluate the performance of the system. They are,

- Possible zones (max 3 zones out of the total 9 zones)
- 1st ranked zone with error of 1-zone distance
- 1st ranked zone with no error

After the user submitted a location query, the system



**Figure 9** Comparison between connectivity of the graph and possible zones.



**Figure 8** Floor plan with the locations of the access points and zones.

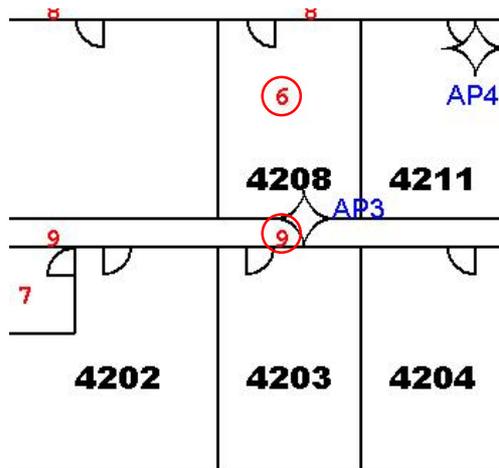
will return a list of possible zones for the user’s possible locations. A ranking will be given to the zones to indicate their relative likelihoods to be the correct location of the user’s location. The neighboring zones of the 1st ranked zones are the 1st ranked zones with error of 1-zone distance. Table 4 shows the results of the 8 experiments. We can easily see that in all of the tests WHAM! identified the correct zone among the top three zones it returned (first case) with the top zone identified being no more than one zone off the actual location (second case).

In the third case, two out of the eight test cases were wrong, but both were due to the same mistake made in distinguishing Zone 7 and Zone 9. Figure 10 is a magnified version of the floor plan which shows only the area around Zone 7 and Zone 9.

As shown in the floor plan, Zone 7 and Zone 9 are very

**Table 4**  
THE RESULTS OF THE EXPERIMENTS

Results	Accuracy
1) Possible Zones (max 3 zones out of 9)	100%
2) 1 <sup>st</sup> Ranked zone with error of 1-zone distance	100%
3) 1 <sup>st</sup> Ranked zone with no errors	75%



**Figure 10** Magnified floor plan showing Zone 6 and Zone 9.

close together. Further, the two zones are reachable from each other through two doors that are almost next to each other. This means that if the user is in Zone 7 and the system is unable to distinguish Zone 7 and Zone 9, backtracking will bring the system back to Zone 9, which is not different enough from Zone 7 to disambiguate the two zones. To make the situation worse, the two zones are only separated by a very thin wall, which can not reduce the signal strength in Zone 7 effectively, thus making the signal strength readings of Zone 7 and Zone 9 very similar.

To handle this error, we analyze the collected signal strengths at these two zones. It is found that the min-max ranges of zone 7 were covered by the min-max ranges of zone 9. Thus, if the location was determined to be zone 7, zone 9 was returned as the result too. We want to find a significant access point which is able to distinguish them. From the data, we observed that there is a large difference between the max values of these two zones from AP3 because AP3 is nearest to the two zones and zone 9 is in effect in direct line of sight to AP3. Thus, we set a threshold value for AP3 to correct the error. Although in the experiment manual fine-tuning was performed after observing the error, the fact that Zone 7 and Zone 9 are hard to distinguish could have been discovered from a reachability analysis on the floor model [4] and special treatment could have been done to handle zones that are direct reachable without corners or thick walls to change the signal strengths between them..

By backtracking on the signal strength data, more accurate location estimation can be deduced. This can be illustrated from the example below. As shown in Figure 10, zones 6 and 9 are very close together and they are only separated by a very thin wall. If only the final signal

strength data were used to determine the location, the two zones are very difficult to distinguish using only WiFi signal strengths. However, if backtracked signal strength data are also taken into consideration, Zone 6 and Zone 9 can be distinguished, because they are not connected together by any corridors or doors.

Compared to the method proposed by Ho. et. al. [3], our method does not require user training and our experiments were conducted in a compact area consisting of dense and contiguous locations. Ho. et. al. reported 80%-85% accuracy with two training sets where only three possible locations were used. Although the accuracy as reported in Table 4 is only 75%, the errors cases were due to two closely connected locations, which, as discussed, could be identified and corrected by analyzing the location model. It is fair to conclude that our method, without the need for training, is more practical and more robust in a real environment.

## VII. CONCLUSION

In this paper, we presented WHAM!, which is a WiFi-based localization method. WHAM! is unique in that it uses the signal strength values recorded along the user's path and a floor model of the area encoding the zones and reachability of the zones to deduce the user's current location. The system was set up in an area of dense and contiguous locations. The accuracy was shown to be 75% accuracy and the erroneous cases were all due to the same error source, which was caused by two rooms separated by a thin wall and reachable from each other. By analyzing the floor model and fine-tuning the threshold, this consistent error can be easily eliminated.

In the current design of WHAM!, only very primitive statistical methods were used. However, we have successfully demonstrated that the signal strength data collected from the user's previous locations is very useful for enhancing the accuracy and robustness of localization.

## ACKNOWLEDGMENT

The authors would like to thank Kenneth Leung and Nic Kong for their helps in this project.

## REFERENCES

- [1] P. Bahl and V. Padmanabhan, RADAR: an in-building RF-based user location and tracking system. *Proceedings of the Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, Vol. 2, 2000.

- [2] Gaetano Borriello. Place lab. Available: <http://placelab.org/>, 2006.
- [3] W. Ho, A. Smailagic, D. P. Siewiorek and C. Faloutsos An Adaptive Two-Phase Approach to WiFi Location Sensing. *Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMMW 06)*, 2006, 452-456.
- [4] H. Hu, and D.L. Lee. Semantic Location Modeling for Location Navigation in Mobile Environment. *Proceedings of the 5th International Conference on Mobile Data Management (MDM'04)*, Berkeley, USA, Jan 2004, 52-61.
- [5] J.H. Kang, W. Welbourne, B. Stewart, and G. Borriello Extracting Places from Traces of Locations. *ACM SIGMOBILE Mobile Computing and Communication Review*, Vol. 9, No. 3, July 2005, 58-68.
- [6] F. Lassabe, P. Canalda, P. Chatonnay, D. Charlet. Refining WiFi Indoor Positioning Renders Pertinent Deploying Location-Based Multimedia Guide. *Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA 06)*, 2006, 126-132.
- [7] Z. Xiang, S. Song, J. Chen, H. Wang, J. Juang, and X. Gao. A Wireless LAN-based Indoor Positioning Technology. *IBM Journal of Research and Development*, Vol. 48, No. 5/6, 2004.