International Global Navigation Satellite Systems Society
IGNSS Symposium 2013

Outrigger Gold Coast, Australia
16-18 July, 2013

A Comparison of algorithms adopted in fingerprinting indoor positioning systems

Zhao Kai
Li Binghao
Andrew Dempster
School of Surveying and Spatial Information Systems University of New South Wales
Sydney, Australia
+61 424420022 kai.zhao@student.unsw.edu.au
Chen Lina
School of Computer science Zhejiang Normal University
Jinhua, China

ABSTRACT

Fingerprinting technology has been widely used in indoor positioning systems such as Wi-Fi positioning systems. Its performance depends on not only the measurement of signal strength, but also the algorithm used. In this paper, an overview is given of the current popular algorithms adopted in Wi-Fi indoor positioning system, including deterministic method (K nearest neighbour, K weight nearest neighbour), probabilistic method and neural network. In order to get a reliable and representative result, those algorithms are evaluated based on the same database. Comparisons were made with respect to positioning accuracy, computational complexity and the size of the database. Furthermore, details of choosing parameters and implementing of these algorithms are discussed.

KEYWORDS: indoor positioning, fingerprinting method, algorithm

1. Introduction

Fingerprinting technology is a well-known method to improve the precision of wifi-based indoor positioning. Because of the complexity of indoor environment, it is hard to model the radio propagation. Thus, in fingerprinting method, a database is generated which contains the features collected on a range of test points. In the positioning phase, users can compute their most likely location by comparing their observed data with the database. Meticulous measurements can insure the positioning results have a high accuracy. At the same time, algorithm used in positioning procedures is also an important factor. It determines the final precision of the result, robustness of the system and the complexity of the positioning calculation. This paper presents an overview of common algorithms which are proposed to be used in indoor positioning systems. An experiment has been conducted to compare their performance in a specific environment.

2. Positioning algorithm

In fingerprinting positioning method, there are at least 5 algorithms which are often mentioned: probabilistic method, K-nearest-neighbor, neural networks, support vector machine (SVM) and small M-vertex polygon (SMP).

2.1 Probabilistic method

This method commonly considers positioning as a classification problem. Assuming that there are n reference points: L_1, L_2, $L_3 \ldots L_n$ and s is the observed signal strength vector. Define $P((L_i|s))$ as the probability of the user in location L_i, the decision rule can be obtained:

Choose L_i, if $P((L_i|s)) > P((L_j|s))$, for j = 1,2…,n, j≠i (1)

According to Bays rule, it is known that $P((L_i|s))=P((s|L_i))\ P(s)/P(L_i)$. If assuming all the test points have equal probability to be accessed and there is no prior knowledge about whether they are available ($P(L_i)$.= $P(L_j)$. for i, j = 1,2,3,..n), the decision rule can be transformed to:

Choose L_i, if $P((s|L_i)) > P((s|L_j))$, for j = 1,2…,n, j≠i (2)

Commonly, the probabilities are calculated by histogram approach. However, in order to insure the accuracy of probability, it requires a mass of data to build the histogram. Some researchers try to model the distribution of signal strength on reference points and draw a relatively accurate distribution curve with less data. A related research shows that about 30% of the distribution of signal strength can have 2 peaks.

In addition, the decision rule is only capable to discrete reference points while users' location could be anywhere in an accessible environment, so, an improved approach suggests to calculate the probability at each reference points, choose 3 or 4 most likely candidates and calculate their weighted average as the estimation of the user' location.

2.2 K-nearest- neighbour

In this method, k reference points in the fingerprinting database are chosen, which are the closest to the test point by measuring the RSSI distance. Then, the test point's location can be estimated by averaging the coordinates of reference points. The RSSI distance usually defined as Euclidean distance or Manhattan distance in signal space and it often was consider as the weight of each reference point.

The principle of this algorithm is relatively simple and the parameter *k* is adjustable for better performance. Binghao Li (2005) proposed that *k* can be adjusted in same environment according to the distance in signal strength space. When there is more than *k* reference points are all close enough to test point in signal strength space, their coordinates should be all considered while calculating location of test point. Otherwise, less than *k* reference points are chosen.

2.2.1 KWNN algorithm with variance counteracted

In classic KWNN (K weighted nearest neighbor) algorithm. The distance in RSSI space is utilized as the weight of each candidate reference point. Because it assumes that the distance in RSSI space is proportional to the geometric distance.

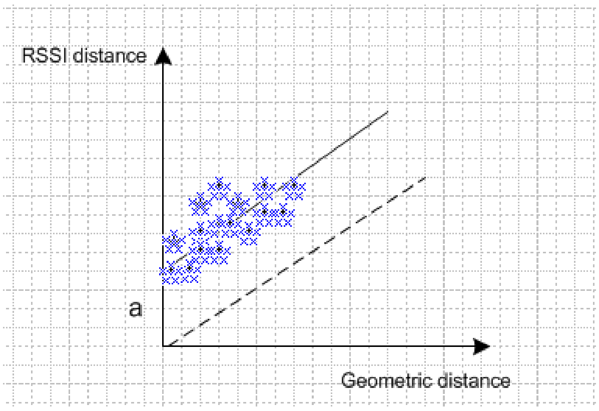

Figure 1 a schematic diagram of the relationship between RSSI distance and Geometric distance

However, the relationship between these 2 distances is not certain because of the signal fluctuation. Generally, the points are distributed around a line when the test point is near to the reference point. More importantly, the line would not pass the origin, it means that when the test point coincides with the reference point, the average of measured distance in RSSI space is not zero. Thus, this test point is more likely to be located among some reference points rather than located on the specific reference point according to the formula of position calculation of KWNN. The bias *a* (shows in Figure 1's coordinate) is related to the standard deviation of the samples at such reference point. The expectation of its square can be calculated in following formula:

$$E(d_{RSSI\ TP=RP}^{2}) = \sum_{i=1}^{n} \sigma_i^2 \tag{3}$$

n is the number of accessed APs, σ_i^2 is the variance of one AP's signal strength at such reference point.

Theoretically, it is hard to determine its own expectation without knowing the distribution of wifi signal strength. However, we can subtract same amount of variance while computing the RSSI distance. Thus, the relationship between 2 distances can be more similar to a proportional relationship (As the dotted line shows). Thus the positioning result should be more accurate.

2.3 Neural network

In this method, a neural network is used. The data of signal strength vectors collected at reference points is used to train the neural network. The network can be described by a matrix W.

$$\boldsymbol{W} = [w_1, \cdots, w_n, w_{n+1}]^{\mathrm{T}} \tag{4}$$

Matrix W contains vectors of operation parameters for each reference point. The function of each vector $\boldsymbol{w}_i$ is to identify whether the input signal strength vector X is yielded at its corresponding reference point.

$$y_j = f(\boldsymbol{w}_j^{\mathrm{T}}\boldsymbol{X}) = \begin{cases} +1, & \text{if } \boldsymbol{X} \in \omega_j \\ -1, & \text{if } \boldsymbol{X} \notin \omega_j \end{cases} \quad 1 \le j \le M \tag{5}$$

In training phase, the value of $\boldsymbol{w}_i$ should be amended according to the result of identify those vectors in database, whose location is known. The aim of the training phase is to determine the matrix W, insuring that the output of each training data can match its right location. When the training phase finished, users just need to compute the vector of signal strength with the matrix w and it would be classify user to a reference point. This method proceeds the positioning as a classification task and the result of positioning is discrete. In order to reduce the complexity of computation and enhance the system's robustness, the structure of the network has been improved. Multi-layer network is adopted, for example, Miao Kehua (2011) used to have a test to adopt BP-network in indoor positioning.

2.4 Support vector machine (SVP)

Support vector machine is a new and promising technique for data classification and machine learning. According to Hui Liu's (2007) introduction, this method has been used successfully in wireless fingerprinting. This method also has two phases: training and positioning. In training phase, it maps those training data to a high-dimensional characteristic space, and then determines a hyper plane between different groups of data. The hyper plane must maximize the distance from the data of both sides. In positioning phase, we can classify the user's location by the judge function:

$$f(x) = \omega^{T}\varphi(x) + b \tag{6}$$

While ω is the weight of the hyper plane and b is the threshold. There are many reference points and it requires more than one judge function to classify the user's location to its nearest reference point. Commonly, we calculate the hyper planes between each reference point and the rest and try each related function as well. Another solution is to determine more than one hyper plane in the same high-dimensional characteristic space and set different thresholds for adjacent groups. This solution requests less functions but more complex.

2.5 Smallest M-Vertex Polygon (SMP)

Smallest M-Vertex Polygon is a positioning method which request some communication between users and transmitters (or anchors) emplaced in the test environment. In this method, users send a request to a total of M transmitters and form an M-Vertex Polygon according to their replies. Once we get enough numbers of M-Vertex Polygons, we can choose the smallest polygon and estimate the position by averaging the coordinates of vertices.

3. Experiment

3.1Test bed

The test bed of experiment was located on the 4th floor of Electrical Engineering Building at UNSW (University of New South Wales), Sydney. The test bed includes most of the public areas include corridor, computer lab, stuff room and toilet. The layout of test bed is indicated in Figure 2 and its area is about 330 square meters. As the figure shows, the red crosses represent the reference points and test points are collected separately. There are 68 reference points in total. The determination of coordinate of reference points is a factor which has effect to the accuracy of positioning. The coordinates was estimated according to the map. The precision of the map may cause centimeter-level errors and the measurement by experimenter can bring same level of errors. However, it is acceptable because the precision of indoor positioning is in meter-level.

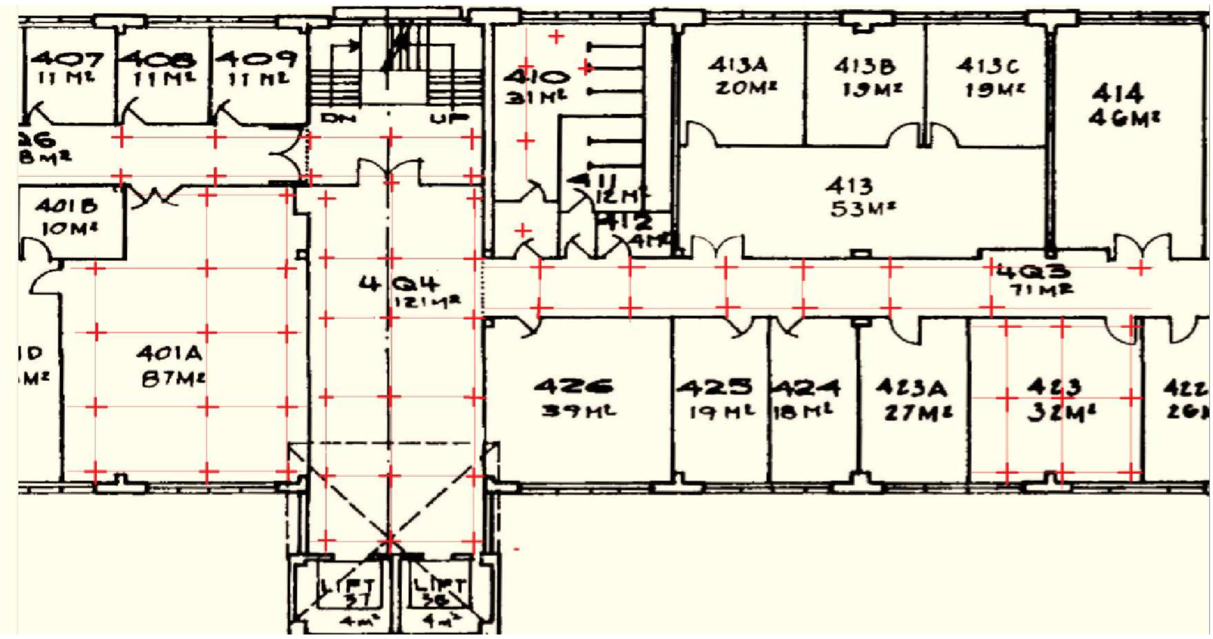

Figure 2 the test bed of this experiment

A common laptop was used to collect the data of signal strength in test bed. Its network adopter is Intel 82579LM which support 802.11n. 802.11n can be adopted at either 2.4GHz or 5GHz and it is compatible with other proposal. Thus, more access points can be adopted in test. This test simulates a low cost system with least pre-deployed devices, so we actually did not deploy any extra APs. All the APs we tracked are all previously deployed by school or laboratory for common use. However, there is a problem that many MAC addresses share a same physical access point and their distributions of signal strength are similar. Since the access points are not deployed by experimenter, it is hard to separate them by physical access point. In this test, we consider each MAC address as an independent physical access point. Although the weight of a physical access point can be increased if it has many MAC addresses, the weight of an access points in calculation is depends on their geometric distribution in test bed. If their location is unknown, technically, it is hard to determine their weight. Thus, the only way is to consider the weight of all the APs are equal in current situation.

3.2 Signal fluctuation

Another important issue is signal fluctuation. Actually, positioning precision is significantly influenced by it. Here is a figure shows the curve of signal strength while receiver is immobile. Its value can fluctuate for 20dBs. Since there are so many APs to track, it sometimes happen that receiver loss a MAC address (considered as an independent access point) in a few seconds while its signal is existent. This can cause a huge uncertainty while positioning. It is also hard to get the character of signal strength at each reference point in the training phase.

Signal fluctuation is affected by many factors. Channel is the one we often consider first. In indoor environment, there is Non-line-of-sight propagation (NLOS) and the change of environment and noise in such frequency can cause fluctuation. The movement of experimenter and other people are included. The second factor is the quality of receiver. A good receiver can return to a more precise result. However, the receiver of a popularized product (e.g. a Smartphone) can not be very expensive and its performance is extremely limited. The third one is signal interference. For example, wireless LAN works on an ISM frequency bandwidth. Other system like Bluetooth, ZigBee works on a same frequency bandwidth. Technically, their codes are orthogonal with each other, but they would add noise to each other. It is not very sure that whether it can affect their signal strength measurement. If the indoor positioning system works on an additional bandwidth, this would not be a problem.

However, in current experiment, there is no solution to reduce signal fluctuation without changing our assumed premise. The NLOS and the movement of human beings must be considered; the wireless network adopter is settled. A paper proposed to filter the signal either in positioning phase (ChaoLin Wu etc. 2004), but it not conform to our assumption, because we simulate that the users can get their real time location while moving. Adopting filter means that the calculation relies on the former data and user have to stay for a while to get their location.

Moreover, the direction of antenna also affects the received signal strength. However, it is hard to determine the direction without compass. Thus, in this experiment, we simply assume the isotropy of all direction and consider all the data collected from the same reference point are equal.

3.3 Filter of Access Points

Another noticeable issue is that there are hundreds of MAC addresses which we considered as Access points. Theoretically, the more APs the better the precision it can achieve. However, the signal strength of some APs is too weak or not stable, and so few samples of RSSI of these APs can be collected. They may cause errors if count them in. Thus, it is required to filter out such APs. In this experiment, we simply choose those MAC addresses which have official names (For example, ‘uniwide’ and ‘SNAP’). Among them, we set a threshold of minimum data amount. Those MAC addresses which have fewer samples than the threshold will be filtered out. The threshold is adjustable. There is a test to evaluate how to set such value to filter most of the unstable APs while keeping most of the useful ones.

3.4 Smoothing the histogram of probability distribution

Generally, the more samples are collected, more accurate the probability distribution can be. However, it is a time-consuming work to generate a fingerprinting database. The amount of samples is limited. In this experiment, there are no more than 100 samples for each reference point. The specific numbers for each access point depends on its own situation. It could be much less than 100. Thus, it seems that the Histogram made of these samples can not coincide with the real distribution very well. Sometimes, a gap may appear at the value which is supposed to have a high probability. Thus, it requires an approach to reduce such error. The approach adopted in this experiment is smoothing the histogram by utilizing the correlation of nearby histogram. The probability of any values would be recomputed with the nearby value considered. In this way, sensitivity of database declined while it can avoid outliers.

4. Test result

This section will present the positioning result and analyse the computational complexity and memory consumption of different algorithms. In our test, the first three algorithms are mainly evaluated. Support Vector machine would be discussed according to others’ research. Smallest M-Vertex Polygon requires a communication between user and the access points, which requires a specific system. Such algorithm has a relatively different application, so it is not included in experiment.

4.2 Positioning result

Figure 3 presents the result of KWNN. The curve shows the trend of average error while the threshold is changing. As can be seen, the error of nearest neighbour is the highest among those algorithms. However, it declined sharply when the threshold is around 35. Theoretically, it is because the data of weak APs are filtered out. However, the robustness is relatively better when K>1, because they just need to find a few nearby reference points and the order of those points is less important. All curves drastically fluctuated when the threshold is around 35. It seems that because there is a number of APs have such amount of samples. Many APs were filtered out. Another noticeable part is that NN algorithm is actually the most accurate algorithm when threshold is higher than 35. However, the difference is small (Except 2WNN, there could be outliers).

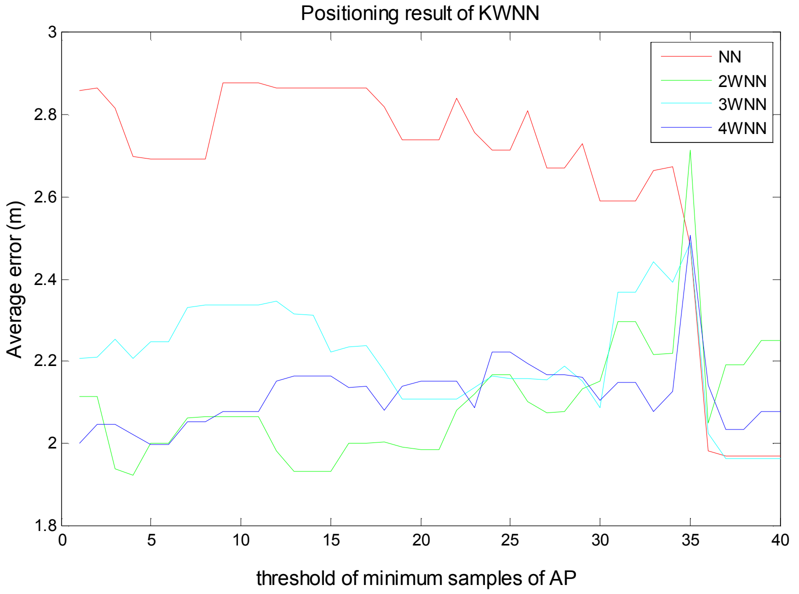

Figure 3 The Positioning result of KWNN

Figure 4 presents a comparison of classic KWNN algrithm and variance counteracted algorithm. It can be seen that the variance counteracted algorithm has a lower average error on most part of the curve. However, it is an average result. According to the analysis of the result, it show that the positioning result of amended KWNN algrithm are all closer to the nearest reference point in RSSI space. However, sometimes this reference point may not the nearest one in real distance. The positioning result could be away from the right position. Thus, there should be a prerequisite for the application of this improved algorithm. It must insure that the measurement is relatively accurate and the user can find the right nearest neighbor in most cases.

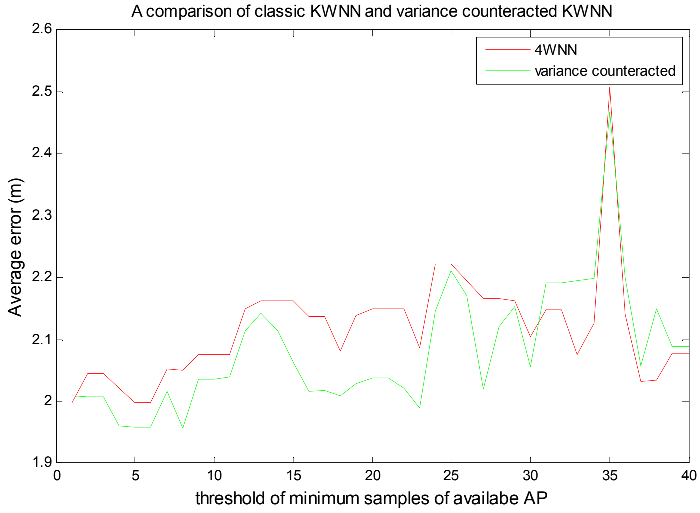

Figure 4 The positioning result of classic KWNN and variance counteracted KWNN

Figure 5 shows the result of different probabilistic methods. Red line represents the classic probabilistic method. Green line represents the method which utilizes the probability as weight and estimates the user's position by averaging the coordinates of the candidate reference points. Yellow line shows the result of the method which has its histogram smoothed. As can be seen, all the methods have a high error when weak APs are kept. It means that those APs with very few samples can cause outliers in probabilistic method. In addition, the weighted method seems has no advantage over the classic method according to the result. However, the accuracy of smoothed method is much better than rest of them, which means that this approach is useful when the number of samples is limited.

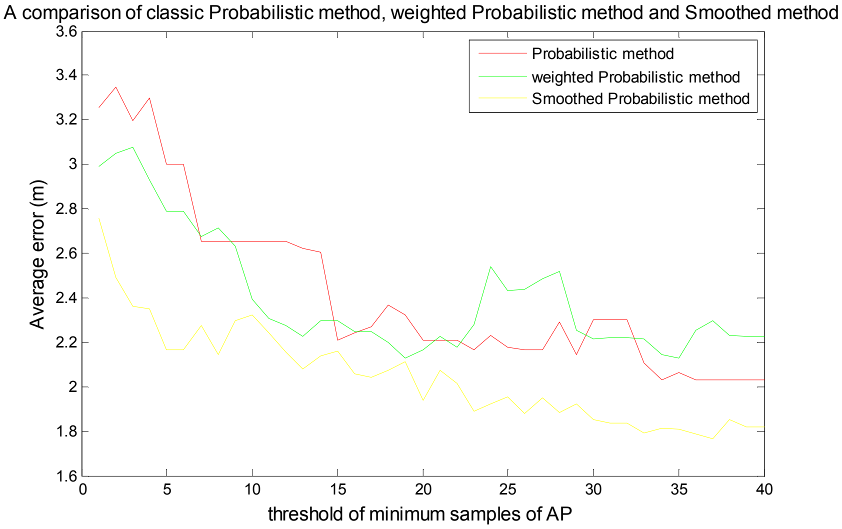

Figure 5 positioning result of probabilistic methods

The average error of neural network is 4.93m, which is quite larger than any other algorithms. According to (Miao Kehua etc. 2011), the accuracy of neural network can be approximately 2 meters (1.34m and 1.67m for x and y axis respectively). There are 2 main differences from this experiment to Miao's. One is that the test bed size, the test bed of this experiment is ten times bigger than Miao's. Technically, it is harder to converge for the matrix of network. Actually, the code must circulate for more than500 times to insure that all the training vectors are classified right. More importantly, the more reference points, the harder it is to divide them by matrix *w.* Another difference is that this experiment adopts basic neural network while BP- network is adopted in Miao's test. Generally, BP-network is more powerful and converges fast. However it has more complex structure and also faces the local minima problem.

4.2 Computational complexity and memory consumption

According to the matlab code for this experiment, the computational complexity of KWNN and probabilistic method is on the same level. However, the sizes of the database are different. KWNN technically just requires the mean value (sometimes the variance) of every AP's RSSI at every reference point. On the contrary, probabilistic method request probability distributions of every AP at a range of reference points. It is approximately100 times bigger than the database of KWNN.

Neural network generally request more times on training phase. Circulation is usually required to achieve an accurate network. In this experiment, the training phase request almost half an hour while the other two algorithms can generate its database in 1minute. However, the advantage of neural network is that once the network is established, it is not required to traverse the database in positioning phase. The positioning computation is much quicker. According to Mauro Brunato and Roberto Battiti (2004), the algorithm of support vector machine have similar feature. As a classification algorithm, it also requires a lot of time on training phase while the computation in positioning phase is simple.

5. Conclusion

In fingerprinting indoor positioning system, the positioning algorithm not only determines the accuracy of positioning result, but also affects the ways to generate the fingerprinting database. In general, the probabilistic method has highest accuracy when adopting the smoothing approach. The average error can be reduced to about 1.8 meters. KWNN is relatively more robust. Its average error is always lower than 3 meters. In addition, KWNN is also employable for interpolated database while probabilistic method can not be adopted.

Classification methods like neural network are turns out to be not suitable for a large test bed. The computational complexity would rise sharply when the number reference points increase. A potential solution is to divide the test bed into pieces and adopts the classification algorithm respectively. Such work will be done in the future.

6. Reference

B. Li, Y. Wang, H. K. Lee, AG. Dempster, C. Rizos, (2005) "A new method for yielding a database of location fingerprints in WLAN," lEE Proc. Communications, 152(5): 580-586.

C. L. Wu, L. C. Fu, and F. L. Lian,(2004) "WLAN location determination in ehome via support vector classification," in Proc. IEEE Int. Conf. Netw. Sens. Control, 2004, vol. 2, pp. 1026–1031.

Hui Liu, Houshang Darabi, Pat Banerjee, Jing Liu, (2007), "Survey of Wireless Indoor Positioning Techniques and Systems", IEEE transactions on systems, man, and cybernetics-Part C: Applications and reviews, vol.37 No.6, November 2007 1067

Miao Kehua, Chen Yaodong, Miao Xiao,(2011) "An Indoor Positioning Technology Based on GA-BP Neural Network", The 6th International Conference on Computer Science & Education August 3-5, 2011

Mauro Brunato, Roberto Battiti, (2004) "Statistical Learning Theory for Location Fingerprinting in Wireless LANs", Elsevier Science 12 October 2004

M. Cypriani, Ph. Canalda, F. Spies (2011), “Performance Evaluation of a Self Calibrated 3D Wi-Fi Fingerprinting Positioning System”, International Conference on Indoor Positioning and Indoor Navigation

V. Kecman, (2001) “Learning and Soft Computing”. Cambridge, MA: MIT Press.

T. Gallagher, B. Li, AG. Dempster, C. Rizos, (2010)"Database updating through user feedback in fingerprinting-based Wi-Fi location systems," in Proceedings of Int. Conf. Ubiquitous Positioning Indoor Navigation & Location Based Service, Kirkkonummi, Finland, 14-15 October,2010, paper 1, session3.