The performance of Simulated Annealing Algorithms for Wi-Fi Localization using Google Indoor Map

Xin Zheng, Guanqun Bao, Ruijun Fu and Kaveh Pahlavan
Center for Wireless Information Network Studies
Worcester Polytechnic Institute
Worcester, MA, 01609, USA
(xzheng, gbao, rjfu, kaveh)@wpi.edu

Abstract - **Wi-Fi localization is currently the most promising approach to build indoor localization systems. Especially after the recent release of Google Indoor Map for the Android system smart phones, many companies such like Skyhook and TCS etc. have flourished into the business of developing accurate Wi-Fi based indoor localization techniques due to their numerous applications. In this paper, we proposed an accurate Wi-Fi based indoor localization algorithm with the help of Google Indoor Map. The initial position of the mobile station (MS) is estimated according to the received signal strength (RSS) from the calibrated Wi-Fi access points (APs). Then, the position of the MS is allocated by using the simulated annealing (SA) algorithm to search for a better solution. During the searching process, the SA takes the indoor map structure into consideration and updates the cost function weights accordingly. This procedure makes sure the final estimation reaches to a better convergence. Extensive experimental results confirm that our solution is able to provide a much better result compared with other existing indoor localization techniques.**

Keywords – Wi-Fi, indoor localization, simulated annealing, Google map

I. INTRODUCTION

In late 1990s, with the rapid growth of wireless communications and mobile applications, the need for location-aware services in indoor environment increased dramatically [1]. A location aware system in indoor environments has the promise of significantly improving the quality of people's life in many ways. For medical applications, nurses could use indoor localization systems to track patients such like elderly and children who are away from visual supervision. It also can be used to assist the sight-impaired patients to locate facilities and other equipment at hospital. For public safety and military applications, indoor localization systems can be used to track prisoners in jail and guide policemen, firefighters, and soldiers in accomplishing their missions inside buildings. More importantly, with the recently release of Google Map for Indoors (shown in Fig. 1), tens of thousands of applications were developed for commercial usage. For example, indoor navigation systems that installed on the portable devices such like smartphones, tablets or PDAs could be used as indoor "GPS" to guide people walking in unfamiliar airports, shopping malls and large grocery stores. People could just launch the Google Maps on their smart phones to find a specific place or discover cool spots nearby. Besides, location information of the smart phone users has a precious advertisement value. Based on the customers' locations, retailers can send out their coupons or sales information to the right person at the right time.

Due to the motivations stated above, many research efforts have been made to devise efficient indoor localization techniques. Existing techniques can be generally classified into two categories: time of arrival (TOA)-based techniques and received signal strength (RSS)-based techniques [2]. The basic idea behind TOA (or TDOA) based technique is to estimate the distance between mobile station (MS) and base station (BS) by multiplying the speed of light with the transmission time and the position of MS can be obtained by Trilateration [3]. However, this type of schemes suffers great errors due to the multipath especially in indoors where the direct path is always attenuated or absent. On the contrast, the RSS based techniques are not that sensitive to the multipath. They take advantage of the well-established Wi-Fi infrastructure inside buildings and calculate the distance between MS and APs based on the intensity of the received signal power. Plenty of RSS based localization algorithms were developed during the past few years. Some commonly used ones include Centroid method [4], K-nearest neighborhood (KNN) [5], AP density method and Gaussian Kernel method [6]. Centroid method, which is the simplest positioning algorithm, estimates the geographic location of MS by averaging the AP positions whose RSS are above a certain threshold [7]. K-nearest neighborhood (KNN) method is a clustering algorithm, it calculates the Euclidean distance between every training nodes and the observation node and the training node with the smallest RSS Euclidean distance value is selected to be the MS position. AP density method describes the overlapped feature of the number of shared APs. The more APs between the MS and the reference node, the more weight this reference node gains. However, all these methods are highly dependent on the deployment density of APs. Thus, in this paper, we adopted the Gaussian Kernel method due to its environmental adaptive characteristic and relative high accuracy. Detailed information about how Gaussian Kernel method works is introduced in section III. After the initial position was estimated by the Gaussian Kernel, a Simulated Annealing (SA) optimization procedure was carried out to refine the estimation result. The SA is an iteration process that simulates the annealing of solids to find the global minimum cost function. Since the indoor map is provided by Google, we adjust the cost function value according to their positions on the map. In this way, map is used as a constraint to guide the searching process. As the iteration goes on, the

978-1-4673-1881-5/12/$31.00 ©2012 IEEE

"temperature" drops down and the search range decreases. Finally, the SA stops when an acceptable good solution is found within a fixed iteration time.

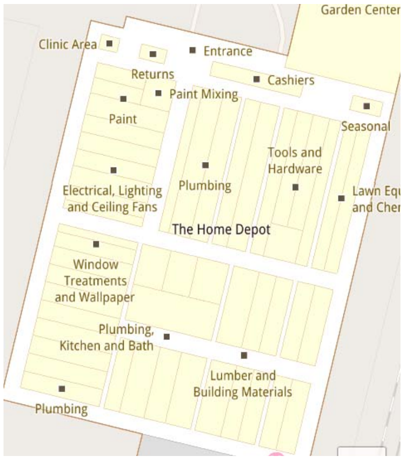

Fig. 1 Google indoor map for Home Depot

The rest of the paper is organized as follows. Section II gives a brief introduction of 802.11 models and how we model the radio propagation channel inside the Home Depot. Sections III describes the detailed methodology that we proposed for the localization. A Gaussian Kernel method is implemented to obtain the initial position, then we emphasis on a SA based optimization procedure to illustrate how Google map can help in searching for a better solution. Extensive experimental results are given in section IV which confirms that the proposed method has a better performance compared with other existing localization methods. Finally, conclusion and future work are addressed in Section V.

II. CHANNEL MODELING

802.11 path loss models are the most commonly used models to approximate the signal attenuation as a function of the distance between transmitters and receivers. The standard way of constructing a channel model is to measure the average received signal strength versus distance step by step. Derived from Friis' equation [8], the general path loss function is stated below:

$$L_p = L_0 + 10\alpha log_{10}(d) \tag{1}$$

where L_p is the path lost (dB), L_0 is the first meter path lost (dB), d is the distance between transmitter and receiver and α is the power-to-distance gradient. Generally speaking, the first a few meters of the channel are always line-of-sight (LOS), while as the distance gets larger, non-line-of-sight (NLOS) takes over. Thus, equation (1) can be further divided into the piecewise functions below:

$$L_p = L_0 + \begin{cases} 10\alpha_1\, log_{10}(d) & ; d < d_{bp} \\ 10\alpha_1\, log_{10}(d_{bp}) + 10\alpha_2\, log_{10}(d/d_{bp}) & ; d > d_{bp} \end{cases} \tag{2}$$

where α_1 and α_2 are the power-distance gradients for different distance ranges respectively. d_{BP} is the breakpoint distance [2].

Another important issue in channel modeling is the shadow fading [8]. Since the received signal strength fluctuates a lot due to scattering, reflection and diffraction of radio waves. We need to add fading margin x to the original equations and this fading margin follows a Gaussian distribution.

$$L_p = L_0 + \begin{cases} 10\alpha_1\, log_{10}(d) + x & ; d < d_{bp} \\ 10\alpha_1\, log_{10}(d_{bp}) + 10\alpha_2\, log_{10}(d/d_{bp}) + x & ; d > d_{bp} \end{cases} \tag{3}$$

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{4}$$

where σ and μ are the variance and mean of the Gaussian distribution respectively.

III. POSITION ESTIMATION ALGORITHM

A. Gaussian Kernel Method

Assume there are a couple of APs which are used as reference points with accurate GPS locations and one MS with unknown position. The MS reads the RSSs from those APs and based on the RSS vector, location of the MS could be estimated by probability distribution of the observation. For any given location l, we know the observation distribution $p(o/l)$. According to the Bayesian rule, the posterior probability of the location can be obtained by:

$$p(l|o) = \frac{p(o|l)p(l)}{p(o)} = \frac{p(o|l)p(l)}{\sum_{\hat{l}\in L} p(o|\hat{l})p(\hat{l})} \tag{5}$$

where $p(l)$ is the prior probability of being at location l, L is the given reference locations set. $p(l|o)$ is called likelihood function which gives the conditional probability of observation o at the given location l. We assume l follows a uniform distribution since $p(o)$ does not depend on the location variable l. Therefore, a simpler posterior probability function can be expressed as below:

$$p(l|o) = \frac{p(o|l)p(l)}{p(o)} = \lambda \cdot p(o|l) \tag{6}$$

where λ is a constant value. Then, the unknown location of MS can be calculated by:

$$p(l|o) = \sum_{\hat{l}\in L} \hat{l} \cdot p(\hat{l}|o) = \lambda \sum_{\hat{l}\in L} \hat{l} \cdot p(\hat{l}|o) \tag{7}$$

In the Kernel method, a probability mass is assigned to a "Kernel" around each of the observations in the training data. Thus, the resulting density estimate for an observation o in location l is a mixture of n_i equally weighted density functions:

$$p(o|l) = \frac{1}{n_i}\sum_{l_i=l} K(o; o_i) \quad (8)$$

$$K_{Gauss}(o; o_i) = \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(o-o_i)^2}{2\sigma^2}\right) \quad (10)$$

where n_i is the number of training vectors in l, $K(o; o_i)$ denotes the Kernel function. σ is an adjustable parameter that determines the width of the Kernel. It has a great effect on the shape of the feature space [9]. Determination of the kernel parameter σ is not an easy task and has been extensively studied in the literature using techniques such as cross validation. Silverman [10] recommends the following formula as a quick estimate of the parameter for a Gaussian Kernel but cautions against over fitting:

$$\sigma^* = (\frac{1}{2k+1})^{1/(k+4)}\hat{\sigma}n^{-1/(4+k)} \quad (11)$$

$$\hat{\sigma}^2 = \frac{1}{k}\sum_{l=1}^{n}\sigma_l^2 \quad (12)$$

where $\hat{\sigma}^2$ is the average of marginal variances; k is the number of APs used for positioning. The above formulas are a quick estimate of the parameter for the Gaussian Kernel.

B. Simulated Annealing algorithm using Google map

SA is an optimization technique that analog the physical annealing process of solid materials [11]. Each iteration of SA algorithm attempts to replace the current solution by a random solution that chosen from its neighborhood. The new solution is accepted with a probability that depends both on the difference between the cost function values and also on a global parameter T (called the temperature), which is gradually decreased during the process. As the temperature drops, the material becomes more rigid and its searching range decreases along with the temperature. The iteration process continues until the temperature reaches the ambient temperature at which the material is perfectly solid and has the lowest energy. To enhance the accuracy of this optimization procedure, we add the Google indoor map into the annealing process. From the Google Indoor Map, we can clearly find the position of the aisles and metal frames in between. Since the customers have a much higher probability walking in aisle than climbing on the metal frames or wandering through the walls (which is obvious), different part of the map has a different bias on the localization result. Based on this observation, we established a mathematic model that adjusts the cost function value according to the MS's observed positions.

The search begins with an initial temperature T_o and ends when the temperature reaches the target temperature T_t. The "temperature" is decreased geometrically by multiplying by a cooling parameter α, where $0 < \alpha < 1$. If the new solution is better than the current solution, it is always accepted without a test [12]. In the algorithm stated below, $l(x, y)$ represents the starting location. $BS(x, y)$, $CS(x, y)$ and $TS(x, y)$ represent the best solution, current solution and the trial solution for the problem respectively. C_{min} is the cost function for the best solution.

The cost function $C(x, y)$ is defined as the summation of Distance Measurement Error (DME) between estimated distance and Euclidean Distances for each reference AP:

$$C(x, y) = \sum_{i=1}^{k} |d_i - D_i| \quad (13)$$

$$D_i = \sqrt{(x-x_i)^2 + (y-y_i)^2} \quad (14)$$

where, k is the number of APs that under consideration, d_i is the estimated distance derived from equation (3), D_i is the Euclidean Distances between the current solution and a selected AP. x_i and y_i are the known coordinates for each AP.

To make the cost function map dependent, we defined a cost function surface $SurfC$, which is a 3D surface that represents the cost function values for every location inside the building:

$$SurfC(x, y) = w_i * C(x, y) \quad (15)$$

where w_i is a weight factor that is adjustable to the structure of Google Indoor Map. For example, if the estimated position locates on the aisle, w_i is a relatively smaller value, which makes the value of the cost function smaller. Oppositely, if the estimated position locates on the metal frames or outside of the building, w_i is a relatively larger value, which makes the value of the cost function larger as well. The detailed Google indoor map assisted SA algorithm for Wi-Fi localization is stated as follows:

step 1. *Initialize the temperature T; Let the initial best solution* $BS(x, y) = l(x, y)$, $C_{min} = SurfC(BS(x, y))$ *current solution* $CS(x, y) = BS(x, y), s = 0$.

step 2. *Generate a trial solution TS by adding (s = 0) or dropping (s = 1) one location from* $CS(x, y)$, *where s is a binary number randomly generated for selecting the adding or dropping operations.*

step 3. *Let* $\Delta C = SurfC(TS(x, y)) - SurfC(CS(x, y))$.

step 4. *If* $\Delta C < 0$, *the trial solution is accepted, set* $CS(x, y) = TS(x, y)$.

Else If $\Delta C >= 0$, *the trial solution is accepted when* $exp(-\Delta C/T) > r$, *where r is a random number in [0, 1]. If the trail solution is accepted, set* $CS(x, y) = TS(x, y)$. *Otherwise, go to Step 2.*

step 5. *If* $SurfC(TS(x, y)) < C_{min}$, *the trial solution is the best solution up to now and set* $BS(x, y) = TS(x, y)$, $C_{min} = SurfC(TS(x, y))$.

step 6. *Repeat Step 2 to 5 for M iterations.*

step 7. $T = \alpha T$.

step 8. *Repeat steps 2 to 7 until* $T < T_t$.

IV. EXPERIMENTAL RESULTS

Since so far Google only released the indoor maps for selective places such like Mall of America, IKEA, Home Depot and Macy's etc., for simplicity reasons, we took our measurement in the local the Home Depot. The Home Depot is a relative open indoor environment, where the roof is very high (about 7-8 meters) and the aisles are separated by metal frames that filled with loads. A snap shot of Home Depot is shown on the left of Fig.2.

Sixteen APs were found by "war driving" inside the building. They were symmetrically distributed on the roof. As shown on the right hand side of Fig. 2, we labeled the APs on the Google Indoor Map, from which we can evaluate the accuracy of different algorithms.

Fig. 2 Indoor scenario of the Home Depot

A. Analysis of channel model

In this section, we demonstrate how we established the channel model. The relationship between the distance and RSS were measured from right under the AP to 75 meters away from the AP. Each set of RSS samples was taken at every 2 meters. Based on the results of those measurements, statistic algorithms were utilized to determine the parameters of the channel model. Fig. 3 shows a scatter plot of RSS versus logarithm distance of the same AP and Fig. 4 shows the corresponding probability density function (PDF) of the shadow fading. Detailed parameters of the channel are illustrated in Table 1.

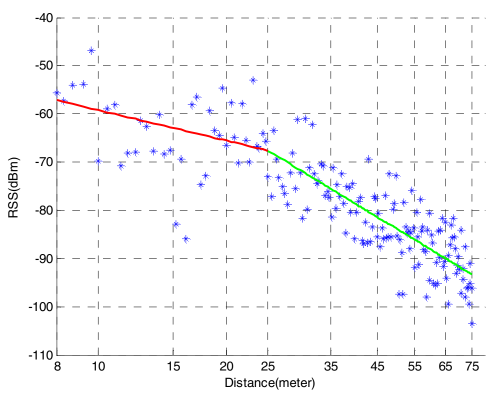

Fig. 3 Path lost model

Table 1. Parameters of the channel

Parameters of the channel	L_0	α_1	α_2	d_{BP}	σ	μ
	38 dB	2.12	3.28	25 m	6.73 m	1.14m

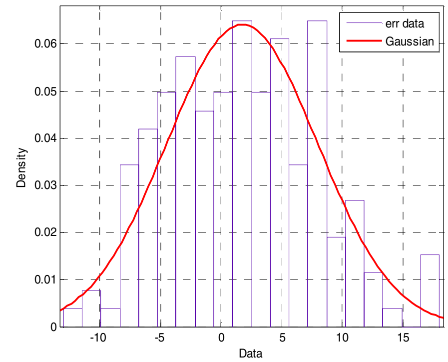

Fig. 4 Shaow fading distribution

B. Simulated Annealing optimize

SA is a heuristic algorithm. It means if the characteristic of the problem is known better, a better performance can be achieved. In this section, we evaluated the performance of the proposed Google Indoor Map assisted SA algorithm using the real data taken from the Home Depot. Based on the real data, we set the parameters: Maximum iteration number, the dimension of the solution space, the Step size and the temperature. Also, we calculate the weight function $w_{i,}$ based on the accuracy, the execution time, the complexity. Fig.5 shows the 3D surface cost function for a sample location. The lowest point of the surface is what we are looking for.

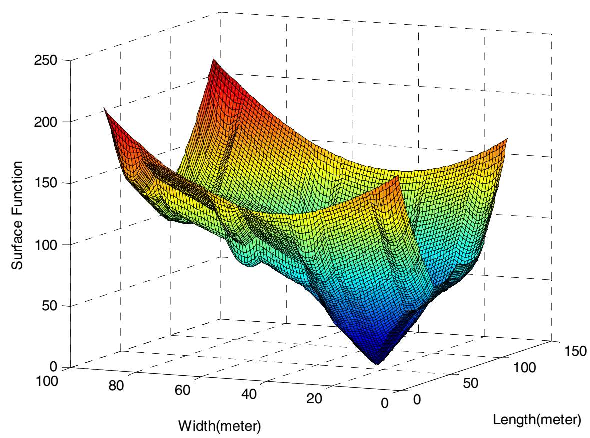

Fig. 5 3D surface of the cost function

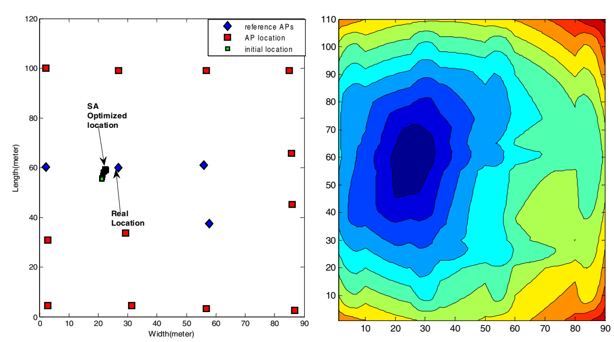

Fig. 6 SA convergence result versus 3D cost fucntion surface

The initial position of the MS is determined by the Gaussian Kernel method, which was detailed in Section III. The parameter $\sigma = 0.01$ is derived from equation (11) and adjusted according to the real data. The SA convergence result versus 3D cost fucntion surface are shown in Fig. 6. There are sixteen APs symmetrical distributed in it. The red and blue points present the location of each AP in the coordinate; the difference is that the blue ones are the reference points for the SA method. The coordinate of the green point is (21.6, 55.9), which is the initial location estimated by the Gaussian Kernel method. The black trail is the process of the SA method; the coordinate of the optimized location is (23.1, 58.4) and the coordinate of real location is (26.9, 61). So the DME reduce from 7.35 to 4.6, from which we can see the improvement of the localization.

C. Performance Evaluation

Finally, we compared the accuracy of the proposed algorithm with other positioning algorithms such as the Google "find my location", Centroid, KNN, AP density and pure Kernel method. It can be seen from Fig. 7 the proposed algorithm has the best performance in terms of Cumulative Distribution Function (CDF). Table 2 provides detailed error statistics for each algorithm.

Table 2 CDF of different localization methods

Methods	Google indoor map	Centroid method	KNN	AP Density method	Pure Kernel method	Improved SA
DME	28.34	20.15	9.91	9.41	10.33	6.79

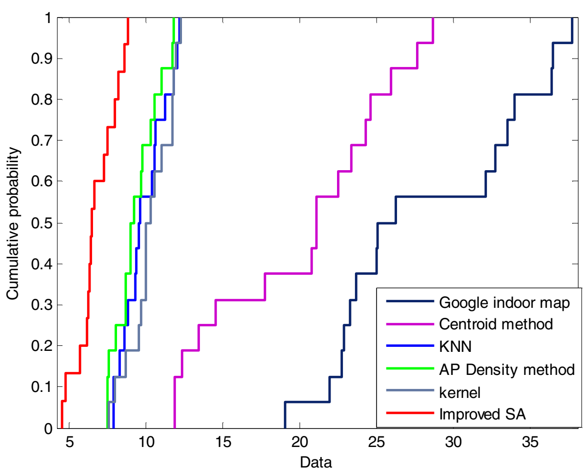

Fig. 7 Performance comparison of different methods

V. CONCLUSION

In this paper, we presented a Google map assisted Wi-Fi localization algorithm for indoor environment. The initial positioning process was implemented by Kernel method. Then, a SA based optimization process is guided by the Google indoor map. The proposed method was tested by using real data from the Home Depot and shown to be more accurate compared with the Google indoor map find your location, centroid method, KNN, AP density and Pure Kernel method.

REFERENCES

[1] K. Pahlavan, F. O. Akgul, M. Heidari, A. Hatami, J. M. Elwell and R. D. Tingley, "Indoor Geo-Location in the Absence of Direct Path", IEEE Wireless Communications, Vol. 13, Issue 6, pp. 50-58, 2006.

[2] K. Pahlavan, X. Li and J.P.Makela, "Indoor Geolocation Science and Technology", IEEE Communications Magazine, Vol. 40, Issue 2, pp. 112 - 118, Feb 2002.

[3] H. L. Groginsky, "Position Estimation Using Only Multiple Simultaneous Range Measurements", IRE Transactions on Aeronautical and Navigational Electronics, Vol. 6, Issue 3, pp. 178-187, 1959.

[4] P. Pivato, L. Palopoli and D. Petri, "Instrumentation and Measurement, IEEE Transactions on", Vol. 60, Issue 10, pp. 3451-3460, Oct. 2011.

[5] T. Bagosi and Z. Baruch, "Indoor localization by Wi-Fi", 2011 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP), pp. 449-452, Aug. 2011.

[6] T. Roos, P. Myllyma,H.Tirri, P.Misikangas, J. Sievnen. "A Probabilistic Approach to WLAN User Location Estimation", International Journal of Wireless Information Network, vol.9, no.3, July 2002.

[7] Y. Cheng, Y. Chawathe, A. LaMarca. "Accuracy Characterization for Metropolitan-scale Wi-Fi Localization", in Proc. MobiSys,2005

[8] K. Pahlavan and P. Krishnamurthy, Principles of Wireless Networks — AUnified Approach, Prentice Hall, 2002.

[9] Azadeh Kushki, Konstantinos N. Plataniotis and Anastasios N. Venetsanopoulos, "Kernel-Based Positioning in Wireless Local Area Networks" IEEE Transaction on mobile computing, vol. 6, no. 6, pp.689-705, June. 2007.

[10] B.W.Silverman, Density Estimation for Statistics and Data Analysis. Chapman and Hall,1986.

[11] Wei-Yu Chiu and Bor-Sen Chen. "A Mixed-Norm Approach Using Simulated Annealing with Changeable Neighborhood for Mobile Location Estimation", IEEE Transaction on mobile computing, vol. 9, no. 5, pp. 633-642, May. 2010.

[12] Zheng.J, Regentova, E. and Varadarajan, R. "Dynamic Planning of Personalized Location Areas for Future PCS Networks with a Simulated Annealing Algorithm". IEEE 63rd Vehicular Technology Conference, vol.2, pp608-612, May. 2006 .