Indoor Localization by WiFi

Timea Bagosi
Free University of Bozen-Bolzano
Email: timea.bagosi@yahoo.com

Zoltan Baruch
Technical University of Cluj-Napoca
Email: Zoltan.Baruch@cs.utcluj.ro

***Abstract*—The demand for accurate indoor localization techniques is increasing. The existing infrastructure of 802.11 WiFi networks can be exploited to position a device in a building. In this paper a positioning system is presented to locate a robot device in the Faculty of Computer Science building. Several techniques are experimented and the algorithms K Nearest-Neighbours (KNN), weighted KNN, weighted Centroid, and Gaussian kernel methods are implemented and compared.**

I. Introduction

Location-aware computing is one of the recent research areas exploiting modern communication technology. Mobility has increased the need for positioning applications, which locate users with the help of their devices. Indoor tracking information can be used to provide context-aware services such as in museums, shops, hospitals, for navigation, guidance and surveillance.

The most popular positioning system working worldwide – GPS – is unsuitable for indoor localization. Finding an adequate alternative for indoor environments is the focus of this research area. Among the numerous technologies, Wireless Local Area Networks (WLAN), using the 802.11b standard (WiFi) provides local wireless access to network architectures. The availability of these networks throughout buildings is increasing, hence providing an existing infrastructure. They can be used for positioning by converting measurements of the received signal strength (RSS) to a distance measurement, and by adding a localization server.

The biggest problem when exploiting the availability of WLAN-s is the signal's instability, which concerns localization accuracy. Propagation of radio signals is complex: range, temporal-, small- and large-scale variations, diffraction, reflection, distortion and scattering are the main obstacles achieving accuracy. Positioning can be conducted only in places where several access points (AP) cover the whole area in consideration, so that a position can be inferred at any place and time.

This paper presents an indoor localization system deployed with a mobile wireless robot in the faculty corridor. Section 2 presents the existing major approaches so far and the main localizing algorithms. Section 3 shows previous research and results in the field. Section 4 proposes the experiments to be conducted. Section 5 presents the test results, and Section 6 concludes the paper.

II. Background

Two major localization approaches exist depending on whether the AP location coordinates are known or not: trilateration and fingerprinting.

Trilateration is the most commonly-known signal propagation approach. Signal propagation models assume an exponential attenuation model for WiFi signals and use the path loss to determine likelihoods based on distance from the AP. In order to use this technique at least three base stations with known coordinates are required. The distance from the stations to the mobile device has to be calculated. The signal strength (SS) measurements should be converted into a distance value with the help of a signal propagation model. A circle with radius equal to this distance from each station indicates the propagation model. The intersection of the circles gives the possible position of the mobile unit.

It is extremely difficult to build a sufficiently good general model of signal propagation that coincides with the real world situation, because indoor radio signal attenuation is almost never radially symmetric, which severely limits the accuracy of these techniques.

A fingerprint is a unique set of AP-s associated to the position, where those access points with certain signal strengths are heard [10]. A database of fingerprints with the known location coordinates are recorded in the training (offline) phase. In the positioning (online) phase, a WiFi device performs a scan of its environment. The algorithm compares this scan with all of the fingerprints in the map to find the one that is the closest match in terms of AP-s seen and their corresponding signal strengths.

A. *Offline calibration phase*

The location determination system uses an a priori intensity map (wireless-map), which captures the environment-specific information needed to reference the location of the mobile device. In any spot the combination of wireless networks heard and their strengths is unique, and has to be measured to build the sample map database. During this training phase a walk-around in the target environment with a mobile device is performed, recording samples of signal strength of multiple access points. The area to be calibrated is typically overlaid with a set of grid points to precisely determine the reference points (coordinates), where sample data should be acquired.

Because of the fading and other disturbances, the observed signal strength at a location varies over time. A compromise has to be made deciding on the grid cell size (the sampling points' density and distance to each other) and the number of sample vectors collected from one point. More reference points, or in other words, smaller granularity, means higher

978-1-4577-1481-8/11/$26.00 ©2011 IEEE

accuracy. Depending on the technique, the actual sample vector can be replaced by the mean signal strength vector of all measurements for the reported coordinates. Furthermore, any changes to the environment that would affect signal strength necessitate re-training.

B. Online positioning phase

The actual positioning phase represents the localizing algorithms, which references the wireless map. A location positioning technique uses the currently observed signal strengths and previously collected information to figure out an estimated location.

There are three main types of positioning algorithms: deterministic, probabilistic, and other methods. Deterministic algorithms attempt to find minimum signal distance between a detected signal strength vector and the location vectors of the various calibration sample points. This may not always be equal to the minimum physical distance between the actual device location and the recorded location of the sample. The resulting sample point of the positioning algorithm is generally regarded as the best raw location estimate. The Nearest Neighbor method (NN), K-Nearest-Neighbor (KNN), centroid and rule-based methods ([2]) fall into this category.

In the Nearest Neighbor method the distance between the observed signal strengths and the recorded ones is computed, in signal space. Therefore a scan of three AP-s with signal strengths $[ss_1, ss_2, ss_3]$ and a matching fingerprint vector $[SS_1, SS_2, SS_3]$ computes the distance with the Eucledian metric as:

$$d = \sqrt{((ss_1 - SS_1)^2 + (ss_2 - SS_2)^2 + (ss_3 - SS_3)^2}. \quad (1)$$

The KNN method takes K nearest neighbors. To obtain the position, the K fingerprints with the least distance to the observed scan are selected, and their location coordinates are averaged. K can be varied, previous study [1] showing that $K = 4$ results in a good accuracy. Similar to the KNN method is the WKNN (weighted KNN) method, which computes the weighted average rather than the simple arithmetic average. In general, KNN and WKNN can achieve better accuracy than the simple NN algorithm.

One of the simplest algorithms, the Centroid, combines all readings for a single access point in the training phase by computing the arithmetic or weighted mean of the positions reported in all the readings. Thus the map contains a single geographic location for each of the APs. With the help of this map, the user is positioned at the center of all of the AP-s heard during a scan, by computing an average of the estimated positions [1].

Although the deterministic methods give reasonable localization accuracy, they discard much of the information present in the training data. Probabilistic algorithms use probability inference to determine the likelihood of a particular location, given a location vector array. The calibration database itself is considered as an a priori conditional probability distribution. Particle filters ([3][4][5]), kernel method, histogram method ([7]), the Hidden Markov Model method ([3]) and many others ([8]) fall under this category.

In the kernel method a probability weight is assigned to the surrounding kernel of each observation in the training data. Thus the resulting density estimate for an observation o in location l is a mixture of n equally weighted density functions, where n is the number of training vectors:

$$P(o; l) = \frac{1}{n} \sum_{i=1,n} K(o; o_i), \quad (2)$$

where $K(o; o_i)$ denotes the kernel function [6].

III. Related Work

Frederic Moster and Ashley Tews describe in [9] a WiFi localization for autonomous industrial vehicles based on particle filters. The environment for the experiment was an industrial work-site divided into 168 square regions of 10 by 10 meters. They built two intensity maps with regions of size 5 by 5 and 10 by 10 meters. The global error of the positioning was within 15 to 30 meters, regardless of the map and region size. In practice, the Bayesian approach proved to localize better than the KNN method. The median error using Gaussian as summarized signal strength of 1.4 meters was achieved in [3].

Yu-Chung Cheng et al. in [1] compared a suite of wireless-radio-based positioning algorithms to understand how they can be adapted for ubiquitous deployment with minimal calibration. Their work proposes a positioning infrastructure to maximize coverage across entire metropolitan areas. It collects data by war driving, a process in which WiFi beacons map the location of access points with the help of WiFi and GPS equipped computers. Data is continuously collected by a variety of users as they move throughout the region. The main advantage is that it can work also outdoors, achieving an accuracy in the range of 13-40 meters.

Teemu Roos et al. in [7] compared the nearest neighbor, kernel and histogram methods. The test area consisted of a one-floor office with normal environmental conditions. The training data was collected twice within 5 days, by using a 2 meter grid, with 20 observations. With a short history the probabilistic methods were more accurate than the nearest neighbor method, while with the full history, the accuracy was approximately the same.

Galo Nuno-Barrau and Jose M. Paez-Borrallo performed in [3] a simulation of 10 different buildings, each with 49 possible locations, 4 AP-s and 10 calibration samples per location. A detailed comparison is conducted between two KNN, two Bayesian and two other algorithms.

IV. Proposed Method

The objective is to implement a localizing system using the Rabbit RCM5400W Core Wireless Module, deployed in the Faculty corridors, Automation and Computer Science Faculty main building's first floor. In the project four different types of algorithms were chosen to serve as the localization performer. The purpose of implementing several algorithms is to conduct a comparison between them, and evaluate the results.

The KNN method searches for the K best matches of the wireless signal strength readings to the wireless map database, and calculates the arithmetic mean of the K positions to localize the mobile terminal. The Weighted K Nearest Neighbor method generates the final result through weighted mean calculation of the K positions. Both KNN and WKNN were implemented with $K = 5$.

The Weighted Centroid method is based on the simple centroid method, only the positions of the AP-s are calculated as the weighted mean of all positions for each AP in place, with the weights being the signal strengths measured at that position. At the localization phase, the signal strengths represent the rate of proximity to the AP centre, this way also to the position.

Gauss kernel method uses the Gauss function to calculate the vicinity between the map samples and the reading. The kernel function is called for each record in the database, which calculates the Gaussian kernel sum of the corresponding AP strengths. The maximum density estimation gives the position. The probability density function is:

$$K(o; o_i) = \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(o - o_i)^2}{2\sigma^2}), \qquad (3)$$

where $K(o; o_i)$ denotes the Gaussian kernel function between the read signal strength and the database sample, σ, the adjustable parameter being set to 0.43, n being the number of AP-s in consideration.

V. Experimental Study

The most important design decision consists of the distribution of the program between the mobile unit and the localizing device. The approach called indirect remote positioning distributes the program in a balanced way, leaving on the mobile terminal just the necessary functions, and placing all other processing parts in a C# program on a computer. So the web server and the localizing server are separated, each performing its own task: the web server provides access to the wireless readings and robot control, the localizing server provides database storage, position estimation and user interface. The main drawback of this design is that the user interface is not accessible to any user with a wireless unit, it can only be seen and controlled on the computer which runs the program. The project uses this design, with the user access only on the localizing server.

In the calibration phase the number of samples gathered in one measuring point were 5, and they were placed 0.5 meters away from each other in length, and 1 meter away in width, as shown on Fig. 1. The calibration phase took about 5 days, and about 7 days to fill in the database.

The mobile device is a robot consisting of two parts: a wireless unit and the motory unit. For the wireless unit the RabbitCore RCM5400W C-Programmable WiFi Core Module is used attached to the core prototyping board. The moving or motory unit is created using the KSR2 Turning Frog Robot Kit. The electric circuit is modified in order to be attachable to the Rabbit module.

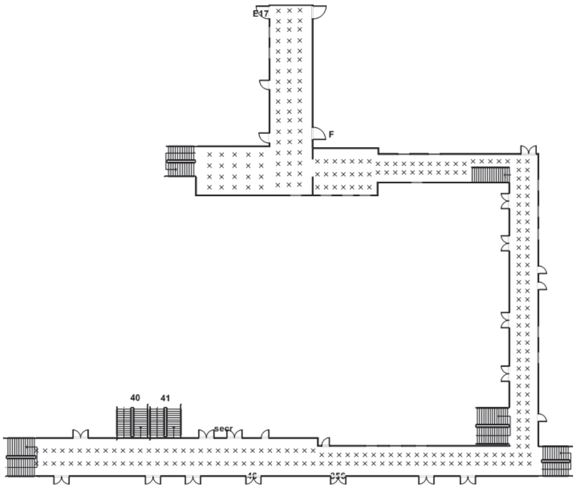

Fig. 1. Corridor map with reading points.

The software part of the implementation is divided in two parts as well: a Dynamic C program loaded on the mobile device, and a C# program stored and run on the localizing server. The Dynamic C program aims to provide motion control and wireless scans during a localization test. The program is a basic controller running a WEB page, and thus the robot acts as a web server. The C# program communicates with the robot and estimates its position.

Two types of tests were conducted: the first test picks randomly some reading value vector from the existing database and feeds it to the algorithms offline. In this case the Gauss algorithm performs best.

The second type of test was performed on-site online. All the algorithms are evaluated using both kinds of tests, with observable difference: error rates highly depend on the environment conditions.

The KNN and the WKNN methods are evaluated together, because of their similarity they should present very small changes compared to each other. Fig. 2 shows the estimation of a position coordinate (400,100) after 5 several runs. About 90% of the estimates obtained by the W5NN method are under

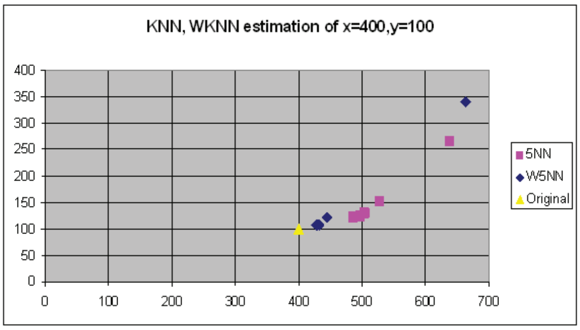

Fig. 2. KNN and WKNN estimation of (400,100).

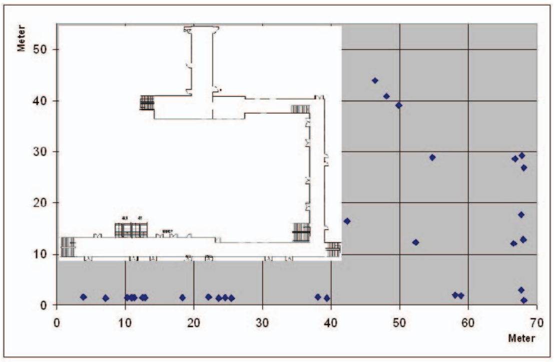

Fig. 3. Centroid estimation of AP centers.

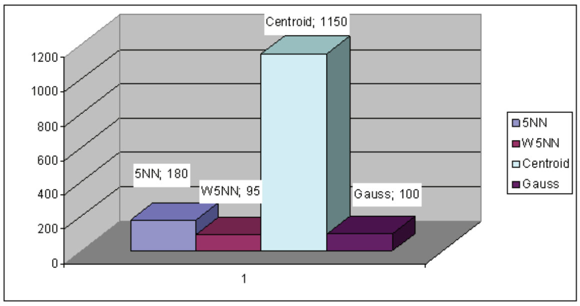

Fig. 5. Error rate in centimeters.

0.5 meter away from the original point, while 90% of the results given by the 5NN are within 1.5 meters. This means that the W5NN method produces 4 out of 5 localizations in the almost exact position as the ground truth, and one with an error of about 2 meters. The 5NN method gives 4 of its 5 positions in the proximity of the original position with the error of about 1.5 meter, and one above 2 meters. In general, the 5NN method shows a greater error rate than W5NN.

Fig.3 shows how the centroid method estimates the AP source locations. Although the centers of the AP-s look mostly in the right position, the algorithm performs very badly. This is due to the fact that the uniform signal propagation model cannot deal with a changing environment.

Gauss kernel algorithm performs perfectly when the first (offline) type of test is performed, and performs with an average error rate of 1.6 meters under the second type of the test. This method proves to provide the highest accuracy in all situations.

Comparing all the algorithms to each other, the following figures can be observed: Fig. 4 shows the proximity of each localized point to the truth value with coordinates (400,100), Fig. 5 shows the error rate in centimeters.

VI. Conclusion

In this paper, we presented the possible indoor localization techniques using wireless networks, we looked at some of the previous experiments, and described our localization system that has been set up for the Computers and Automation Faculty building, first floor corridors. Also, a robot with the necessary navigation and WiFi functions was implemented to serve as a mobile terminal, acting as the web server. A localization system was implemented on a laptop, making use of the collected wireless database map. The application displays the graphical interface with the map and the location of the robot, and with buttons for the necessary control functions: robot control, WiFi reading and localizing algorithms functions.

Several experiments have been conducted to compare the performance of the following algorithms: the weighted Centroid method resulted with the lowest accuracy, the 5-KNN and the 5-WKNN methods showed and average error rate of 1.5 m, while the Gaussian method outperformed all the others with error rate values under 1 meter.

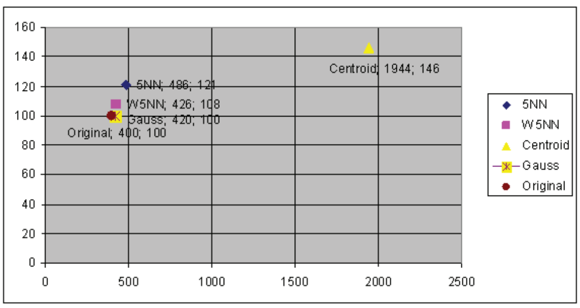

Fig. 4. Overall comparison to (400,100).

References

[1] Yu-Chung Cheng, Yatin Chawathe, John Krumm *Accuracy Characterization for Metropolitan-scale Wi-Fi Localization*. MobiSys 2005m Proceedings of the 3rd international conference on Mobile systems, applications, and services. ACM New York, NY, USA

[2] Wang-Chien Lee, Quixia Chen, Dik-Lun Lee *Rule-based WiFi Localization Methods*. 2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, vol. 1, pp.252-258,.

[3] Galo Nuno-Barrau, Jose M. Paez-Borrallo *A New Location Estimation System for Wireless Networks Based on Linear Discriminand Functions and Hidden Markov Models*. EURASIP Journal on Applied Signal Processing, Vol. 2006, Pages 1-17.

[4] Zhe Chen *Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond*. Statistics 2003, pp. 1-69.

[5] Sajid M. Siddiqi, Guarav S. Sukhatme, Andrew Howard *Experimetns in Monte-Carlo Localization using WiFi Signal Strength*. Proceedings of the International Conference on Advanced Robotics (ICAR), 2003.

[6] Jeffery Junfeng Pan, James T. Kwok, Qiang Yang, Yiqiang Chen, *Accurate and Low-cost Location Estimation Using Kernels*. Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05), Edinburgh, Scotland, July 2005.

[7] Teemu Roos, Petri Myllymaki, Henry Tirri, Pauli Misikangas, Juha Sievanen *A Probabilistic Approach to WLAN User Location Estimation* International Journal of Wireless Information Networks, Vol. 9, No. 3, July 2002

[8] Sinno Jialin Pan, James T. Kwok, Quiang Yang, Jeffrey Junfeng Pan *Adaptive Localization in a Dynamic WiFi Environment Through Multi-view Learning* Proceedings of the 22nd AAAI Conference on Artificial Intelligence (AAAI'07) Vancouver, British Columbia, Canada. July 2007. pp.1108-1113.

[9] Frederic Moster, Ashley Tews *Practical WiFi Localization for Autonomous Industrial Vehicles* Proceedings of the 17th IFAC World Congress, 2008, South Korea.

[10] Venkata N. Padmanabhan Paramvir Bahl. *Radar: an in-building rf-based user location and tracking system*. INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings, pp.775 784.