Increasing Indoor Localization Accuracy Using Wifi Signalst

Chris Pak, Alex Huang
Final Project Report for CS 229
Stanford University, Fall 2011-12

December 16, 2011

Abstract

The widespread availability of wireless networks has created an increased interest in harnessing them for other purposes, such as localizing mobile devices. Our particular goal was to be able to use these techniques as well as others in order to better predict the location of a smart phone inside a building. The current state of the art techniques use Gaussian Process Latent Variable Models to solve this problem. However, existing implementations of this technique require ground traces (wifi data collected sequentially) and are limited to signal-rich environments. Due to the constraints of using phones for this, we were restricted to non sequential data. Since this eleminates the possbility of using traditional GP-LVM techniques, we created a locally weighted regression based machine learning algorithm for geographical localization of a mobile device in an indoor environment based on WiFi signal strength data. Our contribution is a novel approach to generating adequate likelihood models based on locally weighted sums of Gaussian probability densities. The results show that our algorithm is effectively deduces the original floor plan of the environment and presents a more accurate estimate of the device's location.

1 introduction

There has long been interest in the ability to determine the physical location of a device given only WiFi signal strength. This problem, called WiFi localization, has important applications in activity recognition, robotics, and surveillance. This is an extremely challenging task that hes been researched for over a decade now. Recent advances in mobile computing technology and the increasing availability of WiFi networks have enabled more accurate localization in indoor environments where GPS is less precise. Our goal in particular was to find a way to localize something ubiquitous in today's society: the cell phone.

2 Description of Problem

There are several key challenges associated with localization, and in particular doing so with the limitations imposed by using smart phone data. These include data collection, power usage of the phones, and the challenge of localization itself. All three of these aspects add to the difficulty in this problem.

2.1 Data Collection

One obstacle to cell phone localization is in data collection. It is certainly desirable to take thousands of scans of a particular building, and map it to a ground truth map; however, the time and cost of this method is prohibitively expensive. Instead, an intelligent method for extracting the most information with the least cost must be found. Instead of this, we chose to use existing smart phone applications to assist us in data collection. Working with a company that is developing other location based software, we would piggyback data that we would need along with theirs. This means that anyone with this application installed will be collecting data, and will not drain batteries more than would ordinarily occur. While this certainly makes it simple to collect huge amounts of data in a relatively short amount of time, it creates another problem: which data do we need for a particular building. This problem was beyond the scope of our current goals, and for this project, data was simply narrowed down by hand to a specific building as necessary.

2.2 Phone Limitations

Another obstacle to designing an algorithm for smart phones are the limitations of the phones themselves. As mentioned above, one major limitation for phones is battery life. This is in fact the single largest obstacle to designing more complex algorithms, as we cannot insist that the user constantly be scanning wifi networks in order to localize, as that would be an enourmous strain on battery consumption. One aspect of phones that we could not account for at all was dealing with the wide variety of phones and corresponding hardware, putting limits on how much we could trust the data given to us. Finally, the computing and memory requirements have to be taken into consideration. While it is true that smart phones are highly capable machines, the users themselves don't want an application that takes gigabytes of data just to improve accuracy in localization. Similarly, in order for localization to be useful, it must be relatively fast (ie ¡ 1-2 seconds). This also limits how we are able to process incoming data.

2.3 Localization Challenges

The key challenge of localization is overcoming the unpredictability of WiFi signal propagation through indoor environments. The data distribution may vary based on changes in temperature and humidity, as well as the position of moving obstacles, such as people walking throughout the building. This uncertainty makes it difficult to generate accurate estimates of signal strength measurements. Thus, the bulk of research in this area focuses on refining the location likelihood models from the available data collected in the environment.

2.4 Approach

In order to resolve these challenges, we present a machine learning based algorithm for localizing a mobile device using a locally weighted regression to map high-dimensional data to a likelihood function in a low-dimensional latent space. In our context, the high-dimensional data represents signal strength for all WiFi access points in the indoor environment, and the low-dimensional space corresponds to the geographical coordinates of the device's location. Our technique considers that signal strength correlates with physical location. Observations with similar signal strength measurements are likely to be close to each other. This constraint is important when dealing with data from close loops, where the person visits a location at two different points in time. The data we collected consists of signal strength measurements annotated with GPS readings and error estimates. We collected data sets via the method described above, and chose Packard building to model, as it has a relatively simple layout and we had the most data for this building. For calculation, we offloaded this to the server for two reasons in particular. The first, and most obvious, is that local regression is non-parametric and thus we would need to send a phone large amounts of data in order to process it onboard. Second, this allows us to use the significantly higher processing power of the servers.

3 Results

After paring down data for a specific building (Packard in this case) we get a design matrix of 427 observations with a total of 191 detected access points. We denote X as our design matrix, with each row x_n being a vector in $\mathbb{Z}^191$ with each dimension representing the signal strength of each access point (AP) detected by at least one observation. We also have the labels, Y, which represent the estimated GPS location given. Each row of Y, y_n contains latitude, longitude, and the error estimate given by the phone. From here we can actually generate a likelihood function. The standard equation is:

$$J(\theta) = \sum_{i=1}^{N} w_i (y_i - \theta^T x_i)^2 \tag{1}$$

We use the standard weight function,

$$w_i = exp\left(\frac{-\|x_i - x\|_2}{2\tau^2} \right) \tag{2}$$

However, instead of performing a least squares regression, we treat each y_i as a guassian RV with

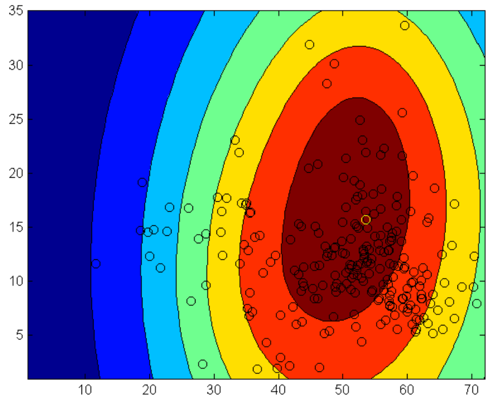

Figure 1: The original GPS readings are black points. For the selected observation, shown in yellow, we generated a likelihood function, shown in a contour plot.

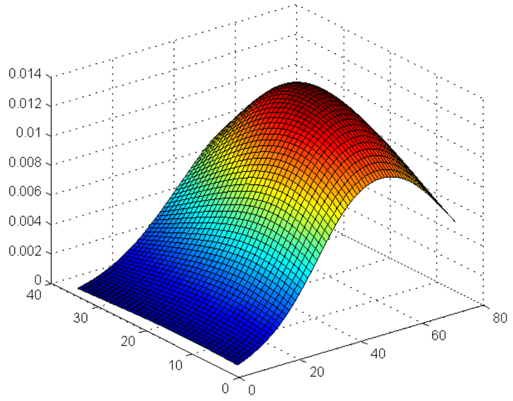

Figure 2: The likelihood function, plotted over the area of the environment.

mean at it's location, and a variance proportional to it's error estimate. We simplify calculations by converting the PDF of each y_i to be in terms of r:

$$P_{y_i}(r) \propto exp\left(\frac{\|y - y_i\|^2}{error^2}\right)/error^2 \qquad (3)$$

This yields our location estimate y as:

$$J(y) = \sum_{i=1}^{N} w_i * P_{y_i}(\|y_i - y\|) \qquad (4)$$

While this function is certainly not convex, we find that in practice, this function is well behaved and we can thus easily find maximum estimates of it. For each observation, we generated a likelihood

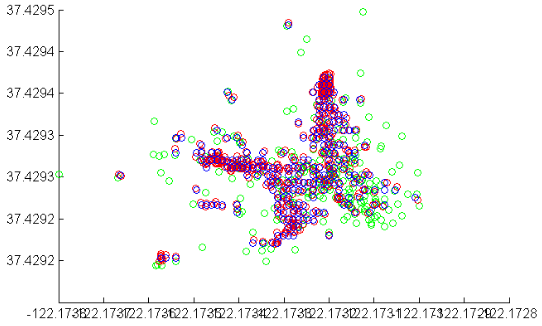

Figure 3: Figure 3: Maximum likelihood estimates (blue), contrasting with the original data (green). We added a small noise component (red) to the blue points because many of the estimates were overlapping.

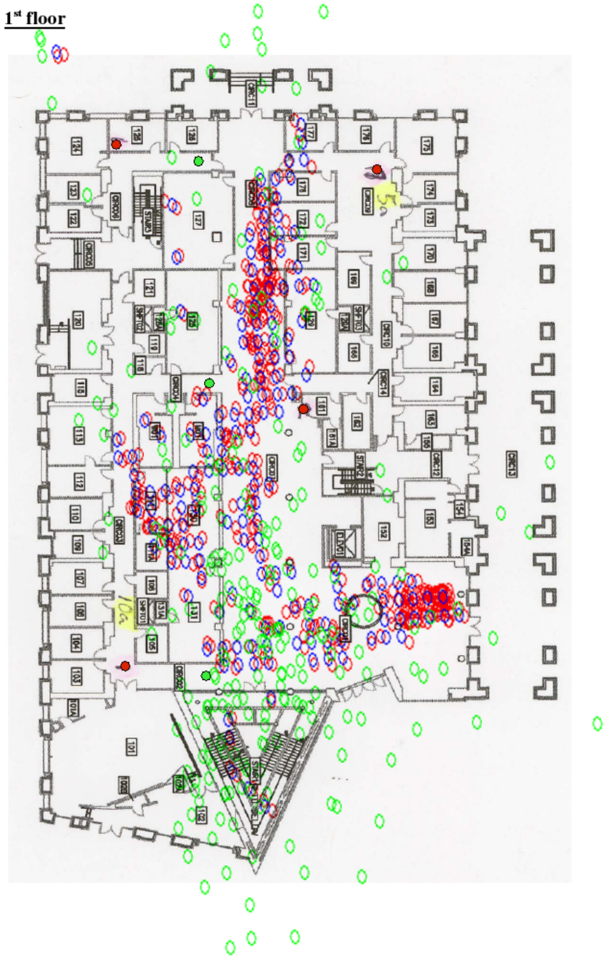

Figure 4: The same data points, overlaid on top of Packard's first floor.

function using this equation. We then chose the maximum value over the distribution to be our MLE for the coordinates of the chosen observation. Iterating over the entire data set, we generated a rough plot of the building's layout. For a data set of 427 observations of 191 access points, our algorithm had a run time of about 10 to 20 seconds. While this certainly seems slow, in fact only one observation will have to be run for a given person, which means that for a given observation, we can have a computation time of less than a tenth of a second, an acceptable speed for the given application.

4 Comparison of localization techniques

Throughout this project we tried several different techniques to get more accurate localization using WiFi. The two that are representative of these methods are Gaussian Process Latent Variable Models and WiFi triangulation.

4.1 GP-LVM

The currently most popular and arguably most state of the art technique is the Gaussian Process Latent variable model. In this model, they use a similar kernel to our in terms of w, a squared exponential weighting for WiFi points. However, they also assign to each WiFi reading a latent variable tied to the time the sample was taken. This allows them to make the following assumptions: successive positions in time have a proximity component, the change in position is likely parallel to other similar reading transitions, and orientation between successive points is relatively low. (this follows from the fact that most hallways are straight corridors. Their final constraint, again like ours, is to assume that similar signal strength readings possess similar coordinates in space. While we began our testing using this model, it was not very effective, since we did not have the temporal data that we needed. Data collection for a single phone occurred at most once every 15 seconds, and more commonly was on the order of several minutes. This essentially zeroed out any calculations involving the latent variables, making this method ineffective.

4.2 Triangulation

This is certainly the most naive approach when attempting localization. Essentially, one considers signal strengths to be distances from an Access point with an exponentially decaying signal strength and random noise factor, ε. When we tried this, we used two steps: We began by using a portion of the data to try to localize the APs nearby, and from there used the rest of the data in for trying to find any error terms. This was even less effective than the above method, due to the faulty assumption that signal strength decays uniformly as a function of radius. As mentioned previously, walls, furniture, and even other people in thte building can effect the signal strength of a particular access point, which makes it virtually impossible to use this method without significant error.

5 Conclusions

We developed an algorithm that predicts latent geographical coordinates based on observed WiFi data by generating likelihood models based on locally weighted sums of Gaussian distributions centered at historical observations. Our algorithm was able to deduce the general structure of the Packard's layout, including the long hallway and the large lobby area. From these results we observed, we conclude that the algorithm can accurately localize a mobile device. Further testing with various bandwidths and threshold levels will be needed to find a better tradeoff between bias and variance. In future studies, we hope to include timestamps into the design matrix and develop constraints based on time data.