Large-Scale Localization from Wireless Signal Strength

Julia Letchner
University of Washington
Seattle, WA

Dieter Fox
University of Washington
Seattle, WA

Anthony LaMarca
Intel Research Seattle
Seattle, WA

Abstract

Knowledge of the physical locations of mobile devices such as laptops or PDA's is becoming increasingly important with the rise of location-based services such as specialized web search, navigation, and social network applications; furthermore, location information is a key foundation for high-level activity inferencing. In this paper we propose a novel technique for accurately estimating the locations of mobile devices and their wearers from wireless signal strengths. Our technique estimates time-varying device locations on a spatial connectivity graph whose outdoor edges correspond to streets and whose indoor edges represent hallways, staircases, elevators, *etc*. Use of a hierarchical Bayesian framework for learning a signal strength sensor model allows us not only to achieve higher accuracy than existing approaches, but to overcome many of their limitations. In particular, our technique is able to (1) seamlessly integrate new access points into the model, (2) make use of negative information (not detecting an access point), and (3) bootstrap a sensor model from sparse training data. Experiments demonstrate various properties of our system.

Introduction

In recent years, the problem of estimating a person's location has gained interest in several research communities. The centrality of location information to such tasks as activity recognition, surveillance, and context-aware computing can be seen in many applications. For example, AT&T, Google, and Microsoft all offer city-scale services including specialized web search, navigation and nearby-friend-finding. In the context of activity recognition, home rehabilitation of people suffering from traumatic brain injuires (Salazar *et al.* 2000) could be supported by the ability to monitor patient movements. On an indoor scale, (Nguyen *et al.* 2003) are able recognize complex behaviors of people by analyzing their motion trajectories.

The goal of our research is to develop a location estimation system that is large-scale and long-term; this broad goal implies several desirable properties:

Indoor & Outdoor Coverage: Many location-aware applications would benefit from full coverage of a person's daily movements; unfortunately, existing location-estimation systems (including GPS) cover *either* indoor *or* outdoor locations, but not both.

Minimum calibration: It is not feasible to collect accurately labeled training data for every location in a large-scale coverage area; therefore, a location system should be able to bootstrap from sparse training data and improve its model using unlabeled data collected through normal system use, over time. Furthermore, it should be able to integrate new signals (e.g. WiFi access points or RFID beacons) automatically, without recalibration or manual adjustments.

Minimum hardware requirements: To minimize barriers to adoption, users should not be required to carry special hardware, such as cameras, in order to use a system. Furthermore, the high cost of installing environment-embedded hardware over large areas should be avoided.

Privacy-observant: While a certain loss of privacy is acceptable in some applications, most users are not willing to be "tracked by their environment".

The final two concerns can be addressed by building a location-estimation system around pre-existing wireless networking infrastructures such as 802.11 access points (APs) and GSM cell towers. Such approaches have indeed been developed, and they infer location from the strengths of wireless signals measured by a user's laptop, PDA, or cellphone (Seidel & Rappaport 1992; Bahl & Padmanabhan 2000; Haeberlen *et al.* 2004; LaMarca *et al.* 2005).

Unfortunately, none of the existing WiFi location techniques meet the remaining requirements. Most of them work only indoors and require extensive training data (Ladd *et al.* 2002; Haeberlen *et al.* 2004) or additional information such as the locations of walls and furniture (Seidel & Rappaport 1992; Bahl & Padmanabhan 2000). While systems such as Active Campus and Place Lab provide outdoor location estimates using little or no calibration by leveraging the widespread deployment of wireless technology in residential areas (LaMarca *et al.* 2005), the accuracy of these systems is too coarse for indoor use. Lastly, none of the existing approaches are able to incorporate new APs as they appear over time, or to use unlabeled data to improve existing AP sensor models.

In this paper, we introduce a novel approach for large-scale, long-term, WiFi-based location estimation. Our technique enables accurate location estimates both indoors and

outdoors, using only small amounts of training data. We use a hierarchical Bayesian sensor model that is refined as more data becomes available, through normal use of the system. The sensor model is integrated into a graph-based representation which can generate accurate location estimates and trajectories on a street map or inside buildings.

We derive our sensor model in the next section. We then describe our graph-based representation and discuss model refinement from unlabeled data. Experimental results are next, followed by conclusions and discussion of future work.

Hierarchical Bayesian Sensor Model

The goal of Bayesian localization is to estimate posteriors over a person's location, x_t, conditioned on all sensor measurements obtained through time t. Our system performs Bayesian localization using a particle filter, which is a technique that represents and propagates such posteriors using sets of weighted samples (Fox *et al.* 2003). Each sample $x_t^{(i)}$ is a potential location of the person, and each has an associated importance weight $w_t^{(i)}$. Standard particle filters realize Bayes filter updates by propagating samples through time according to the following sampling procedure: *Resampling:* Draw with replacement a random sample $x_{t-1}^{(i)}$ from the previous sample set according to the importance weights $w_{t-1}^{(i)}$. *Sampling:* Generate a new particle $x_t^{(j)}$ by sampling from the motion model $p(x_t^{(j)} \mid x_{t-1}^{(i)})$. *Importance sampling:* Weight the sample by the measurement likelihood $p(z_t \mid x_t^{(j)})$.

This section focuses on the sensor model, which is the system component used in the "importance sampling" step to determine the likelihood of observing any particular sensor measurement at any given location. A WiFi sensor measurement consists of a list of APs detected at a particular moment. Each AP detection is annotated with a signal strength, measured in dbs. Existing WiFi sensor models fall into two major classes: signal propagation and fingerprinting. We briefly describe these before introducing our own model.

Signal propagation (SP) models assume an exponential attenuation model for WiFi signals and use this path loss to determine likelihoods based upon distance from the AP, whose location is assumed known (Seidel & Rappaport 1992; Bahl & Padmanabhan 2000). SP models can therefore generalize even to locations for which no training data is available. Furthermore, they require storage only for the location of each AP and a simple description of its signal attenuation (details follow). Unfortunately, attenuation is almost never radially symmetric, which severely limits the accuracy of these techniques.

Fingerprinting (FP) models ignore attenuation and instead compute likelihoods from location-specific statistics compiled from training data. The form of these statistics ranges from raw measurements (Bahl & Padmanabhan 2000) to histograms (Ladd *et al.* 2002) to Gaussian densities (Haeberlen *et al.* 2004), but all FP techniques require far more training data than SP models and do not extrapolate well into areas not covered by this data.

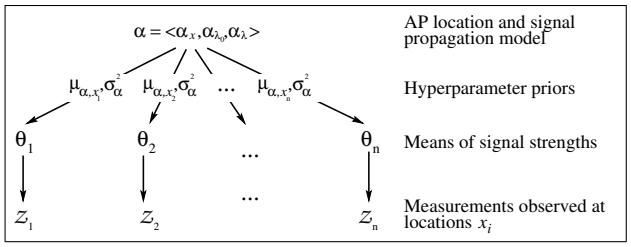

Figure 1: Hierarchical Bayesian model of signal strengths measured at different locations. Measurements are at the bottom level, per-location measurement models are at the middle levels, and the overall model of the AP is estimated at the highest level. The local models are coupled via the priors provided by the AP model.

Overview of the hierarchical Bayesian model

We will now introduce a sensor model that combines the benefits of SP and FP techniques. Since measurements of different APs are independent given the location of the device, we will restrict our attention to the likelihood model of a single AP. Similar to the FP technique of (Haeberlen *et al.* 2004), our approach uses Gaussians to estimate the likelihoods of signal strength measurements at each location. However, instead of performing maximum likelihood estimation independently for each Gaussian, we estimate the Gaussian means using hyperparameters with priors derived from an SP model (Seidel & Rappaport 1992). In essence, these hyperparameters perform spatial smoothing that takes the properties of signal propagation into account.

Figure 1 illustrates our model. The parameters in the model are estimated from training sets $\mathcal{Z}_1, ..., \mathcal{Z}_n$. Each set $\mathcal{Z}_i$ contains the sensor measurements collected at location x_i. In this section, we assume that the measurement locations are known. The lowest level of the model contains a Gaussian likelihood model for each location x_i. All Gaussians share the same variance σ^2, since most measurement variation is due to location-independent sources such as the orientation of the device antenna or nearby cars/people. Key parameters of the model are the means, θ_i, of the local Gaussians. Their values strongly depend on factors such as the distance from the AP and the objects between the device and the AP. The value of each mean θ_i is estimated using Gaussian hyperparameters. The priors $(\mu_{\alpha,x_i}, \sigma_\alpha^2)$ of the hyperparameters at location x_i are extracted from a signal propagation model, the parameters of which are estimated at the highest level.

Likelihood model for *known* AP parameters

We now derive the likelihood of a signal strength measurement z given the device location x and AP parameters $\alpha = \langle \alpha_x, \alpha_{\lambda_0}, \alpha_\lambda \rangle$ (since measurements observed at different locations are independent given the AP parameters, we omit the location index whenever possible). α_x describes the AP's location, and α_{λ_0} and α_λ characterize the attenuation of its signal. Here, α_{λ_0} gives the signal strength measured at a reference distance d_0 from the AP, while α_λ gives the mean path loss exponent modeling the degree to which signal strength decreases with distance from the AP.

The likelihood of signal strengths measured at different locations is modeled by Gaussians with varying means. The

mean θ at a location x is estimated using Gaussian hyperparameters $\langle \mu_{\alpha,x}, \sigma^2_\alpha \rangle$ (see (Gelman *et al.* 2003) for a detailed discussion of hierarchical Bayesian estimation). We apply the SP model introduced by Seidel and Rappaport (Seidel & Rappaport 1992) to generate the prior value for the hyperparameter mean $\mu_{\alpha,x}$:

$$\mu_{\alpha,x} \quad = \quad \alpha_{\lambda_0} - 10\,\alpha_\lambda\, log_{10}\left(\frac{||x-\alpha_x||}{d_0}\right) \qquad (1)$$

As can be seen, this value is a function of the distance $||x - \alpha_x||$ between x and the AP. It falls off logarithmically with distance, at a rate depending on the path loss parameter α_λ. Assuming that the variance σ^2_α is known and independent of location, we get the following prior distribution over the Gaussian mean θ at location x:

$$p(\theta \mid x, \alpha) \quad = \quad \mathcal{N}\left(\theta;\ \mu_{\alpha,x}, \sigma^2_\alpha\right) \qquad (2)$$

The likelihood of observing sensor measurement z at location x now follows from the hierarchical model by integration over the unknown Gaussian mean:

$$\begin{aligned} p(z \mid x, \alpha) \quad &= \quad \int p(z \mid x, \alpha, \theta)\, p(\theta \mid x, \alpha)\, d\theta \\ &= \quad \int \mathcal{N}(z; \theta, \sigma^2)\, \mathcal{N}(\theta; \mu_{\alpha,x}, \sigma^2_\alpha)\, d\theta \qquad (3) \\ &= \quad \mathcal{N}(z;\ \mu_{\alpha,x}, \sigma^2 + \sigma^2_\alpha) \qquad (4) \end{aligned}$$

(3) follows from the fact that the observation is independent of the AP's parameters and location if the parameters of the local Gaussian are known. (4) is a standard convolution of the two Gaussians in the hierarchical model.

Thus far, the likelihood model does not take any training data into account. It can be shown that, given a set $\mathcal{Z}$ of measurements collected at location x, the posterior distribution over the mean θ at location x follows as (compare to (2)):

$$p(\theta \mid x, \alpha, \mathcal{Z}) \quad = \quad \mathcal{N}\left(\theta;\ \widehat{\mu}_{\alpha,x}, \widehat{\sigma}^2_\alpha\right), \qquad (5)$$

where the posterior values of the hyperparameters are:

$$\begin{aligned} \widehat{\mu}_{\alpha,x} \quad &= \quad \frac{m\sigma^2_\alpha}{m\sigma^2_\alpha + \sigma^2}\bar{z} + \frac{\sigma^2}{m\sigma^2_\alpha + \sigma^2}\mu_{\alpha,x} \qquad (6) \\ \frac{1}{\widehat{\sigma}^2_\alpha} \quad &= \quad \frac{1}{\sigma^2_\alpha} + \frac{1}{\bar{\sigma}^2}. \qquad (7) \end{aligned}$$

Here $\bar{z}$ and $\bar{\sigma}^2 = \sigma^2/m$ are the sample mean and variance of the m observations in $\mathcal{Z}$. The updated mean is the weighted average of the prior and data means, while the updated variance shrinks with the amount—independent of the actual *values*—of training data.

Combining (5) with (4), we get the likelihood of observing z given AP parameters α, the device location x, and previously observed training data $\mathcal{Z}$:

$$p(z \mid x, \alpha, \mathcal{Z}) = \mathcal{N}(z;\ \widehat{\mu}_{\alpha,x}, \sigma^2 + \widehat{\sigma}^2_\alpha) \qquad (8)$$

Likelihood model for *unknown* AP parameters

In most applications, neither the location of the APs nor their signal propagation parameters are known. We estimate these values from data collected at all locations. Let $\mathcal{Z}_{1:n}$ denote the sets of measurements observed at the n locations $x_{1:n}$. The probability of a specific parameter vector α for the corresponding AP is then given by:

$$\begin{aligned} p(\alpha \mid \mathcal{Z}_{1:n}, x_{1:n}) \quad &\propto \quad p(\mathcal{Z}_{1:n} \mid x_{1:n}, \alpha) \qquad (9) \\ &= \quad \prod_{i=1}^{n} p(\mathcal{Z}_i \mid x_i, \alpha) \qquad (10) \\ &= \quad \prod_{i=1}^{n} \mathcal{N}(\bar{z}_i;\ \mu_{\alpha,x_i}, \bar{\sigma}^2_i + \sigma^2_\alpha) \qquad (11) \end{aligned}$$

(9) follows by Bayes rule under a uniform prior. (10) leverages measurement independence given the AP parameters. Each Gaussian in (11) computes the likelihood of all measurements in a set $\mathcal{Z}_i$, using the sample mean $\bar{z}_i$ and sample variance $\bar{\sigma}^2_i = \sigma^2/m_i$, where m_i is the number of observations in $\mathcal{Z}_i$ (see (Gelman *et al.* 2003) for a derivation).

We are now prepared to derive the likelihood of a measurement z_i observed at location x_i from an AP with unknown parameters. The general form of this likelihood is obtained by integrating over the AP parameters:

$$\begin{aligned} &p(\ z_i \mid \mathcal{Z}_{1:n}, x_{1:n}) \\ &\quad = \quad \int p(z_i \mid \alpha, \mathcal{Z}_{1:n}, x_{1:n})\, p(\alpha \mid \mathcal{Z}_{1:n}, x_{1:n})\ d\alpha \qquad (12) \\ &\quad = \quad \int p(z_i \mid \alpha, \mathcal{Z}_i, x_i)\, p(\alpha \mid \mathcal{Z}_{1:n}, x_{1:n})\ d\alpha \qquad (13) \end{aligned}$$

The two terms in the integral correspond to (8) and (11), respectively. (13) does not have a closed-form solution, so we approximate the integration by importance sampling from the posterior over the parameter α. We generate different α's from a grid of reasonable AP locations α_x and signal propagation parameters α_λ and α_{λ_0}. The importance weight of each α is then given by (11); using these weights we sample k values α_g at which (13) is then evaluated:

$$\begin{aligned} p(z_i \mid \mathcal{Z}_{1:n}, x_{1:n}) \quad &\approx \quad \frac{1}{k}\sum_{g=1}^{k} p(z_i \mid \alpha_g, \mathcal{Z}_i, x_i) \qquad (14) \\ &= \quad \frac{1}{k}\sum_{g=1}^{k} \mathcal{N}(z_i;\ \widehat{\mu}_{\alpha_g,x_i}, \sigma^2 + \widehat{\sigma}^2_{\alpha_g}) \qquad (15) \end{aligned}$$

The resulting likelihood is a mixture of k Gaussians, with one mixture component for each sampled AP parameter vector. For efficiency, we collapse the mixture at each location into a single Gaussian using the technique described in (Lauritzen 1996).

This finalizes the derivation of our sensor model for signal strength measurements. To summarize, the hierarchical Bayesian technique estimates local Gaussian models using hyperparameters with priors that are estimated using a signal propagation model along with data collected at all locations. More training data available at a specific location generates a more focused Gaussian likelihood. At each location, the model smoothly blends between signal propagation (hyperparameter priors) and fingerprinting approaches (local posteriors), thereby inheriting the benefits of both.

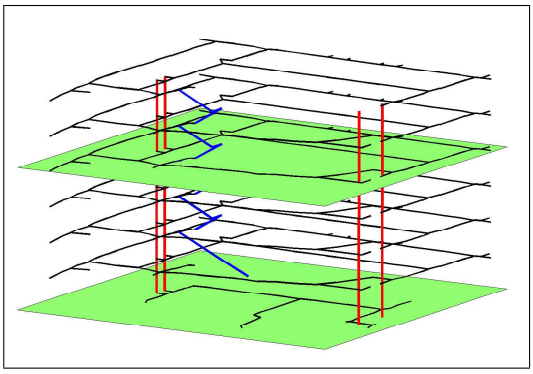

Figure 2: Connectivity graph of our six-story test building (for clarity, basement and fourth floor are shown as shaded planes). Diagonal edges between floors are stairs; vertical edges elevators. The outgoing edges can be connected to an outdoor street graph.

Graph-based Location Estimation System

We will now describe how to integrate the likelihood model into a graph-based particle filter for location estimation.

Graph-based location estimation

Our tracking system is an extension of the approach introduced by Liao and colleagues, in which location is estimated on a spatial connectivity graph (Liao *et al.* 2003; Liao, Fox, & Kautz 2004). Outdoor graph edges correspond to streets or footpaths (for the U.S., such graphs are publicly available on the web), and indoor edges correspond to hallways, staircases, elevators, *etc.* (see Fig. 2). The major advantages of using a graph are its abilities to (1) bias motion models, and (2) simplify sequential location estimates into smooth trajectories.

In graph-based localization, the motion update step of the Bayes filter moves a particle along an edge of the graph. When the particle reaches a vertex, it continues its motion along a randomly-selected outgoing edge connected to that vertex. Motion updates and motion model learning are not the focus of this paper, and we refer the reader to (Liao *et al.* 2003; Liao, Fox, & Kautz 2004) for more information. The measurement likelihood is then computed from the sensor model described in the previous section. Since this model assumes a set of discrete locations, we discretize the edges on the graph and estimate a Gaussian sensor model for each discrete bin.

Learning sensor models from unlabeled data

We will now describe how to learn/improve a sensor model using unlabeled training data (*i.e.*, data without location labels). To do so we assume that an initial, possibly crude, sensor model is available. Such an initial model can either be learned from sparse, labeled data traces, or from a signal propagation model if the locations of APs are known.

Our sensor model derivation assumed knowledge of the locations x_i at which training data sets $\mathcal{Z}_i$ were obtained. To relax this assumption, we use expectation maximization (EM) to simultaneously estimate the sensor model and the locations at which the data was observed. Table 1 illustrates the learning algorithm. Its input is a connectivity graph over

Algorithm Learn_sensor_model $(G, \mathcal{M}, \mathcal{S}, \mathcal{Z})$

1. **Inputs:** Graph structure $G := (V, E)$
   Motion model $\mathcal{M} := p(x' \mid x)$
   Initial sensor model $\mathcal{S} := p(z \mid x)$
   Unlabeled data log $\mathcal{Z} := \{z_1, \ldots, z_N\}$,
2. $k = 0$; $\mathcal{S}_0 = \mathcal{S}$
3. **do**
4. $k = k + 1$
5. $\mathcal{X}_k$ = bayes_filter_smoothing($G, \mathcal{M}, \mathcal{S}_{k-1}, \mathcal{Z}$)
6. $\mathcal{A}_k$ = AP_parameter_samples($\mathcal{X}_k, \mathcal{Z}$)
7. $\mathcal{S}_k$ = sensor_model_posterior($\mathcal{X}_k, \mathcal{A}_k, \mathcal{Z}$)
8. **until** $(\mathcal{S}_k - \mathcal{S}_{k-1}) < \varepsilon$
9. **return** $\mathcal{S}_k$

Table 1. EM-based sensor model learning.

the environment, a motion model, an initial sensor model, and a time-stamped log of sensor measurements. At each iteration (Lines 3–8), the algorithm first performs forward-backward smoothing on the data log using the current sensor model. To do so, distributions over the discretized edges of the graph are extracted from the particles of the forward and backward pass. Multiplication of these distributions gives $\mathcal{X}_k$, a sequence of smoothed location estimates.

For each AP, the smoothed location estimates $\mathcal{X}_k$ are used with the sensor measurements $\mathcal{Z}$ to generate a set $\mathcal{A}_k$ of AP parameter samples. The samples for each AP are drawn according to (11), where the sample mean $\bar{z}_i$ and sample variance $\bar{\sigma}_i^2$ are computed from the expectations generated by the smoothed location estimates in $\mathcal{X}_k$. The AP parameter samples are used with the data log and location estimates to compute posteriors over the Gaussian likelihood models at each location. The prior means for these models are extracted from the AP parameters using (1), and the posteriors are then computed according to (5)–(7). The resulting posterior Gaussians are then collapsed to generate a single Gaussian for each location.

At each iteration of EM, the updated sensor model is used to perform smoothing over the unknown data locations. Typically, these location estimates become more peaked, and the sensor models more focused. EM is stopped as soon as the sensor model does not change significantly.

Using negative information

The sensor model discussed thus far ignores the "negative" information contained in *not detecting* a certain AP. Such an approach is reasonable during tracking, since positive information from several APs is generally enough for localization. This approach also avoids the necessity of reasoning about all non-detected APs, of which there can be thousands.

During learning, however, negative information can be very useful for estimating the AP parameter vector. If an AP is not detected at a given location, then the signal strength at that location is below the detectable range of the mobile device's sensor. To reflect this, we insert a dummy measurment into the data at that location, with signal strength sampled uniformly from values below the device's threshold (determined empirically). This is done for all negative measurements within a certain range of a positive AP dectection. In our tests we found that this approach provides helpful biasing of AP location and parameter estimates in situations

Type	ML mean	ML median	Particle mean
HSM Global	1.8 ± 0.9	0.8 ± 0.1	3.4 ± 0.4
HSM Track	1.2 ± 0.7	0.7 ± 0.1	2.3 ± 0.5
FSM Global	2.1 ± 0.6	1.0 ± 0.1	4.4 ± 0.4
FSM Track	1.8 ± 0.7	0.9 ± 0.1	4.1 ± 0.4

Table 2. Indoor localization error [meters] and 95% confidence intervals using our hierarchical sensor model (HSM) and a flat sensor model (FSM) that uses only local Gaussians.

where the positive data alone is sparse and/or symmetric; see Figure 3, for example.

Experimental Results

In these experiments we evaluate the accuracy of location estimates obtained with our approach, both indoors and outdoors; we also demonstrate that the system can bootstrap from sparse training data. Our data was collected using a standard laptop worn in a backpack by a person walking through a building or driving a car. All experiments used 1,000 particles for localization. Our motion model was a mixture of zero motion (stopping) and a Gaussian velocity with mean at 0.8m indoors and 5.0m outdoors. The on-graph sensor model discretization was 0.5m indoors and 10m outdoors. The same discretizations were used to compute maximum likelihood (ML) location estimates by determining the bin with the highest sum of weighted particles.

Indoor localization This experiment demonstrates the ability of our approach to accurately estimate locations inside large buildings. The 7-floor test environment is represented by the graph shown in Fig. 2. Indoor ground truth was interpolated from a small set of pre-specified waypoints. When our subject reached one of these waypoints, he pressed a button to correlate his location with the current timestamp. These synchronization points were later used to generate ground truth location along the entirety of the trace. We first learned a sparse initial sensor model from a data log annotated with ground truth location, and then used Lines 6 and 7 of the algorithm shown in Table 1 with unlabeled data to learn a refined sensor model.

The top half of Table 2 shows localization accuracy of our model averaged over five test traces, each of which included multiple floor transitions via staircases. The left two columns provide the averages over the mean and median errors of the most likely location estimates during each run. The right column gives the average error per particle. The errors in the "global" row correspond to experiments in which the initial location of the device was unknown. In these tests, the particles typically converged to the correct location (and the correct floor) after less than 15 seconds. The "tracking" row shows results obtained when the first 15 seconds were removed from the evaluations; that is, they show typical tracking errors.

The bottom half of Table 2 shows the estimation accuracy when using the sensor model introduced by (Haeberlen *et al.* 2004). Here, the training data at each location was used to estimate the mean and variance of a flat, non-hierarchical Gaussian sensor model. As can be seen, the accuracy is lower, which is mostly due to the fact that the flat sensor model underestimates the variability of sensor measurements. Furthermore, in contrast to our technique, this flat model provides no means to "extrapolate" into areas in which no training data is available. While a direct comparison to other existing approaches is difficult due to differing representations and environments, our results are consistently more accurate than those reported in the literature, including those based on sophisticated calibration (Bahl & Padmanabhan 2000; Krishnan *et al.* 2004).

Outdoor localization In this experiment, our subject drove a car through the residential area shown in the left panel of Fig. 3. One GPS-annotated data log was used to generate a sensor model, which was then used to localize a user on the test trace shown in the figure. The average localization errors are summarized in the upper row of Table 3.

Model	ML mean	ML median	Particle mean
Full data	15.6	12.3	26.0
Gap data	27.9 ± 13.7	16.1 ± 3.2	39.6 ± 13.1
Unlabeled	16.9 ± 6.7	11.2 ± 0.9	28.1 ± 6.6

Table 3. Outdoor localization error in meters, using sensor models learned from: a complete training set; sets with gaps; and sets with gaps plus additional, unlabeled data.

To assess the ability of our approach to bootstrap a sensor model from sparse training data, we manually removed the training data on one of the six street blocks in the test trace. This resulted in six different test runs, each with a training data gap of one block (up to 200m long). Our hierarchical Bayesian sensor model "filled" these gaps with priors extracted from estimates of the AP parameters. These estimates were generated from measurements obtained outside the gap (Fig. 3, (a)-(c)). The average localization errors are presented in the middle row of Table 3. Not surprisingly, the error increased signficantly relative to results obtained with full training data.

We then added another, unlabeled training trace, and improved the sensor model by running the EM algorithm described in Table 1. After convergence (typically 5 iterations), we evaluated the new sensor model on our test trace (results in the bottom row of Table 3). Note that this accuracy is nearly identical to that of the sensor model learned from full data (top row), indicating that the priors generated by our system are good enough to allow bootstrapping of EM from training data with gaps of 200m. This result is significant because it demonstrates that our approach does not require training data covering the full region of deployment; instead, it can learn from unlabeled data collected by users.

Conclusions and Future Work

Recent research interest in activity recognition and commercial interest in context-aware services have created a strong demand for large-scale, long-term location-estimation techniques. The near-ubiquitous availability of wireless APs in urban areas allows us to leverage WiFi as a location sensor both indoors and outdoors. We have introduced a hierarchical Bayesian technique for learning local Gaussian likelihood models of signal strength. Our approach estimates

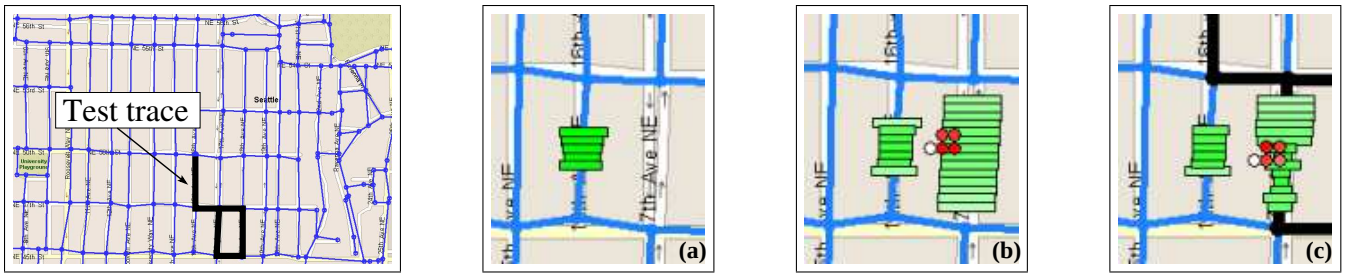

Figure 3: **(Left)**: Residential neighborhood used for outdoor localization. **(a)**: Measurements from one AP in the training data (darker color indicates higher mean; width indicates variance). All data collected on the right, vertical block is manually removed. **(b)**: Sensor model extracted from the data in (a). The gap is "filled" by the prior extracted from the estimated AP parameters. The small circles indicate the five most likely AP locations. Due to the use of negative information, these locations are not symmetric around the AP detections in (a) (the AP was never detected in the left, vertical street). **(c)**: Sensor model from (b) refined using EM with an additional, unlabeled trace (dark edges).

global AP parameters and uses them to generate priors for local Gaussian means. In frequently-visited locations, the local models become more specific and accurate as more data is collected and added to the model.

The sensor model is integrated into a graph-based location estimation system. We showed how EM can be applied to refine the parameters of our sensor model using unlabeled sensor data. The model can be bootstrapped from sparse training data, or from AP parameter estimates. We believe that we have presented the first broad-area location estimation system that can (1) improve its sensor model and add new APs using unlabeled data, (2) work both outdoors and inside multi-story buildings, and (3) leverage negative sensor information. Furthermore, our approach achieves higher accuracy than do existing approaches, while requiring smaller amounts of training data.

The success of our approach indicates that our system would benefit from the relaxation of its current limitations. For example, large open spaces such as parking lots or building lobbies are poorly modeled by connectivity graphs; we are currently devleoping a mixed spatial representation that includes bounded, open spaces inside of which particles can depart from the graph and move freely. Entrances/exits to these spaces will connect to vertices of our standard graph. Additionally, to address the problem of insufficient WiFi density in rural areas, we are exploring the use of GSM cellphone signal strength. We believe that our framework can readily incorporate this information.

Finally, we are currently conducting experiments to bootstrap our sensor model from known AP locations and unlabeled WiFi traces. Initial indoor results indicate that we can achieve median errors of less than 2m. Outdoors, we can leverage public AP location databases such as wigle.net, which currently contains the locations of over 2.4 million APs! These databases are created by WiFi hobbyists who pool their data logs, and can therefore contain inaccuracies; however, we believe that our system's tolerance for uncertainty and ability to improve through EM will allow it to provide accurate location estimates from this data, resulting in a system that requires no additionally labeled data at all.

Acknowledgments

This work was supported in part by the National Science Foundation under contract number IIS0433637, an NDSEG graduate fellowship, and Intel Research.

References

Bahl, P., and Padmanabhan, V. 2000. RADAR: An in-building RF-based user location and tracking system. In *Proc. of IEEE Infocom.*

Fox, D.; Hightower, J.; Liao, L.; Schulz, D.; and Borriello, G. 2003. Bayesian filtering for location estimation. *IEEE Pervasive Computing Magazine* 2(3).

Gelman, A.; Carlin, J. B.; Stern, H. S.; and Rubin, D. B. 2003. *Bayesian Data Analysis*. Chapman and Hall/CRC, 2nd edition.

Haeberlen, A.; Flannery, E.; Ladd, A.; Rudys, A.; Wallach, D.; and Kavraki, L. 2004. Practical robust localization over large-scale 802.11 wireless networks. In *Proc. of the Tenth ACM International Conference on Mobile Computing and Networking*.

Krishnan, P.; Krishnakumar, A.; Ju, W.-H.; Mallows, C.; and Gani, S. 2004. A system for LEASE: Location estimation assisted by stationary emitters for indoor RF wireless networks. In *Proc. of the IEEE Infocom*.

Ladd, A.; Bekris, K.; Rudys, A.; Marceau, G.; Kavraki, L.; and Wallach, D. 2002. Robotics-based location sensing using wireless ethernet. In *Proc. of the Eight ACM International Conference on Mobile Computing and Netwrking (MOBICOM)*.

LaMarca, A.; Chawathe, Y.; Consolvo, S.; Hightower, J. Smith, I.; Scott, J.; Sohn, T.; Howard, J.; Hughes, J.; Potter, F.; Tabert, J.; Powledge, P.; Borriello, G.; and Schilit, B. 2005. Place Lab: Device positioning using radio beacons in the wild. In *International Conference on Pervasive Computing*.

Lauritzen, S. 1996. *Graphical Models*. Oxford University Press.

Liao, L.; Fox, D.; Hightower, J.; Kautz, H.; and Schulz, D. 2003. Voronoi tracking: Location estimation using sparse and noisy sensor data. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Liao, L.; Fox, D.; and Kautz, H. 2004. Learning and inferring transportation routines. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*.

Nguyen, N.; Bui, H.; Venkatesh, S.; and West, G. 2003. Recognising and monitoring high-level behaviours in complex spatial environments. In *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*.

Salazar, A.; Warden, D.; Schwab, K.; Spector, J.; Braverman, S.; Walter, J.; Cole, R.; Rosner, M.; Martin, E.; and Ellenbogen, R. 2000. Cognitive rehabilitation for traumatic brain injury. *Journal of American Medical Association* 283(23).

Seidel, S., and Rappaport, T. 1992. 914 MHz path loss prediction models for indoor wireless communictions in multifloored buildings. *IEEE Transactions on Antennas and Propagation* 40(2).