NOKIA

Research Center
NRC-TR-2010-002

Simple, On-Device Positioning using Compressed Fingerprint Archives

Jonathan Ledlie

Nokia Research Center Cambridge, US
http://research.nokia.com
July 20, 2010

Abstract:

This technical report describes (a) a method for efficient on-device positioning that is based on the radio frequency fingerprint model, (b) a method for compressing the fingerprints that are used with this positioning method and, (c) a method for estimating the physical distance between two or more devices. Our positioning method, called Weighted Gaussian Distance, achieves similar positioning accuracy to the state-of-the-art Bayesian method, but at lower compute costs. Our archival method for compressing large numbers of fingerprints allows large physical areas to be stored compactly on a mobile device, so that offline positioning is possible in most circumstances. The method reduces the size of a typical fingerprint archive by 1-2 orders-of-magnitude, while keeping accuracy roughly equivalent. Both methods can be used in survey-based and organic localization environments.

Index Terms:

Indoor Localization
Lossy Compression
Radio Frequency Fingerprinting
Algorithms

1 Introduction

This technical report describes (a) a method for efficient on-device positioning that is based on the radio frequency fingerprint model [2,3,10], (b) a method for compressing the fingerprints that are used with this positioning method, and (c) a method for estimating the physical distance between two or more devices. Our positioning method, called Weighted Gaussian Distance, achieves similar positioning accuracy to the state-of-the-art Bayesian method [7], but at lower compute costs. Our archival method for compressing large numbers of fingerprints allows large physical areas to be stored compactly on a mobile device, so that offline positioning is possible in most circumstances. The method reduces the size of a typical fingerprint archive by 1-2 orders-of-magnitude, while keeping accuracy roughly equivalent. Both methods can be used in survey-based [2,7,8] and organic [4–6,13] localization environments.

2 Weighted Gaussian Distance Positioning

We propose a new variation on the radio frequency fingerprint method for non-GPS positioning. We call our approach *Weighted Gaussian Distance*, or WGD. The two key advantages to WGD are (1) that it is extremely simple to compute and (2) that it provides a general scan distance function. Because we anticipate localization algorithms running continuously in the background on mobile devices, this simple computation should translate into longer battery life. Scan distance functions are useful for clustering scans. Clustering, in turn, can be used for outlier detection and cleaning scan databases. By themselves, distance functions are also useful for estimating the physical distance between the positions where the scans were made, as discussed in Section 4.

WGD localization works as follows. We begin with the standard space-to-APs histogram common to fingerprint methods (As in Haeberlen [7], we assume spaces and not grid point on a map are the unit of interest.) As previous work has found, these per-space per-AP histograms can typically be summarized as a single Gaussian [7]. When normalized, each histogram can be summarized with mean μ and standard deviation σ. This reduces storage and network transmission requirements to the domain of the histogram (typically $0 - 100$ dBm) times the maximal height of the histogram down to two floating point numbers.

Every space is assigned a fingerprint, which is a set of mappings from access points to data triples:

$$AP_i \Rightarrow \langle w_i, \mu_i, \sigma_i \rangle \tag{1}$$

where w_i is the weight of AP_i, the number of observable APs is τ, and the total weight for each fingerprint $\sum_{i=1}^{\tau} w_i$ is 1. Note that the most recent k scans of the user also form a fingerprint using the same method.

Determining the weight w to apply to each visible AP is an important component of our algorithm. A strawman method would be to simply weigh each visible AP equally: $1/\tau$. Instead, we base the weight on the probability that the given AP will actually be observed in the space. When a space is scanned many times, some APs will be seen in every scan, and some seen only rarely. Intuitively this is because a user's fingerprint will tend to have the same distribution of APs when it is in the same space: if the user's fingerprint does not contain an AP that is almost always observed when in a particular space, it is highly unlikely that the user is in this space. Weighing according to scan detection frequency reflects this intuition. Specifically, the weight is:

$$w_i = \frac{r_i}{R} \tag{2}$$

where r_i is the number of readings of AP_i and R is the total number of readings that constitute this fingerprint, $\sum_{i=1}^{\tau} r_i = R$.

To find the distance between two fingerprints, we determine the similarity in signal strengths of APs that exist in both fingerprints, and penalize for missing APs. The comparison of any two fingerprints returns a distance metric $-1 \leq s \leq 1$, where a comparison of identical fingerprints returns 1 and of disjoint fingerprints returns -1 (Disjoint fingerprints are those that share no access points). For fingerprints A and B, the effect

δ of each AP i is:

$$\delta = \begin{cases} \frac{w_a+w_b}{2} \times O(\mu_a, \sigma_a, \mu_b, \sigma_b) & \text{if } i \in A,\, i \in B, \\ -\frac{w_a}{2} & \text{if } i \in A,\, i \notin B, \\ -\frac{w_b}{2} & \text{if } i \notin A,\, i \in B \end{cases} \tag{3}$$

where $O()$ is the overlap coefficient between the two Gaussian distributions [9]. The total distance is the sum of each δ. Figure 1 provides an example of computing the distance between a pair of fingerprints.

One particularly nice aspect of this overlap computation is that it exists as a closed-form function [9]. Alternatively, the results from the function can be stored in a look-up table [12]; we found a table with only hundreds of values gave almost the same results as a function. This simple computation is in contrast to Bayes, which can require thousands of iterations to converge to a space.

Note that the same comparison applies if space B were a user's fingerprint or if it were any collection of scans.

A special case exists where we have only one RSSI value for an AP. This value becomes the mean μ. Because this situation typically exists for low probability APs, taking more scans is not advisable, because we may need to take many more in order to achieve a stable σ. To estimate σ for these low reading APs, we use a weighted average of this APs σ (if it exists) together with an empirically observed σ_g:

$$\frac{(r_i - 1)\sigma_i + \sigma_g}{r_i} \tag{4}$$

With this, an expected overlap can be computed even with very few RSSI values, or even a single value, from a given AP. We found $\sigma_g = 1$ worked well in our experiments.

Note that the selection of a Gaussian function is orthogonal to the method. While RSSI distributions often do follow a Gaussian and often can be summarized with one, any normalizing function that fits the data would work. In particular, spaces could have their own individual functions (*e.g.,* Beta distribution), which do occasionally fit RSSI data better. The downside to selecting a different function is that the overlap computation (determining the overlap coefficient) might be more computationally expensive – in contrast to the closed-form function that the Gaussians permit.

Note that variations on the weighting function are also possible. For example, not diving by two when an AP is missing. This would serve to penalize missing APs more (and increase the domain of s to -2).

2.1 Evaluating Weighted Gaussian Distance

We evaluated the Weighted Gaussian Distance method in simulation. We use a scan dataset from a nine-story building which contains more than 1,400 distinct spaces. We assigned a fingerprint to each space, assuming knowledge of all scans of the space. For each space, we took eight scan samples, added a small amount of noise to each measurement, built a "user" fingerprint, and then observed which space had the highest matching fingerprint. If the space the localizer guessed was the same as the user's, this was deemed an accurate estimate. We repeated this test 1000 times for each space: for example, an accuracy of 80% means that we correctly localized $\frac{800}{1000}$ times. As these spaces would be excluded by real localizer as having insufficient data, we omitted spaces with fewer than three visible APs or fewer than ten scans, removing 6% of spaces.

Figure 2 (top) shows the effect of weighing according to scan frequency as compared to weighing each AP equally, *i.e.,* setting $w = \frac{1}{\tau}$. While it is possible other refinements exist, such as weighing according to the maximum RSSI value seen for the given AP, it is clear that a reasonable weighting is more accurate than simply valuing all APs equally.

Because the WGD method returns a score for each potential space, it is possible to look down the list of returned spaces beyond the top ranked space. Figure 2 (bottom) shows that the correct space is almost always in the top four ranked spaces, as would be returned to the user. In a visual map application, all of the highly ranked spaces could be highlighted if one did not stand out.

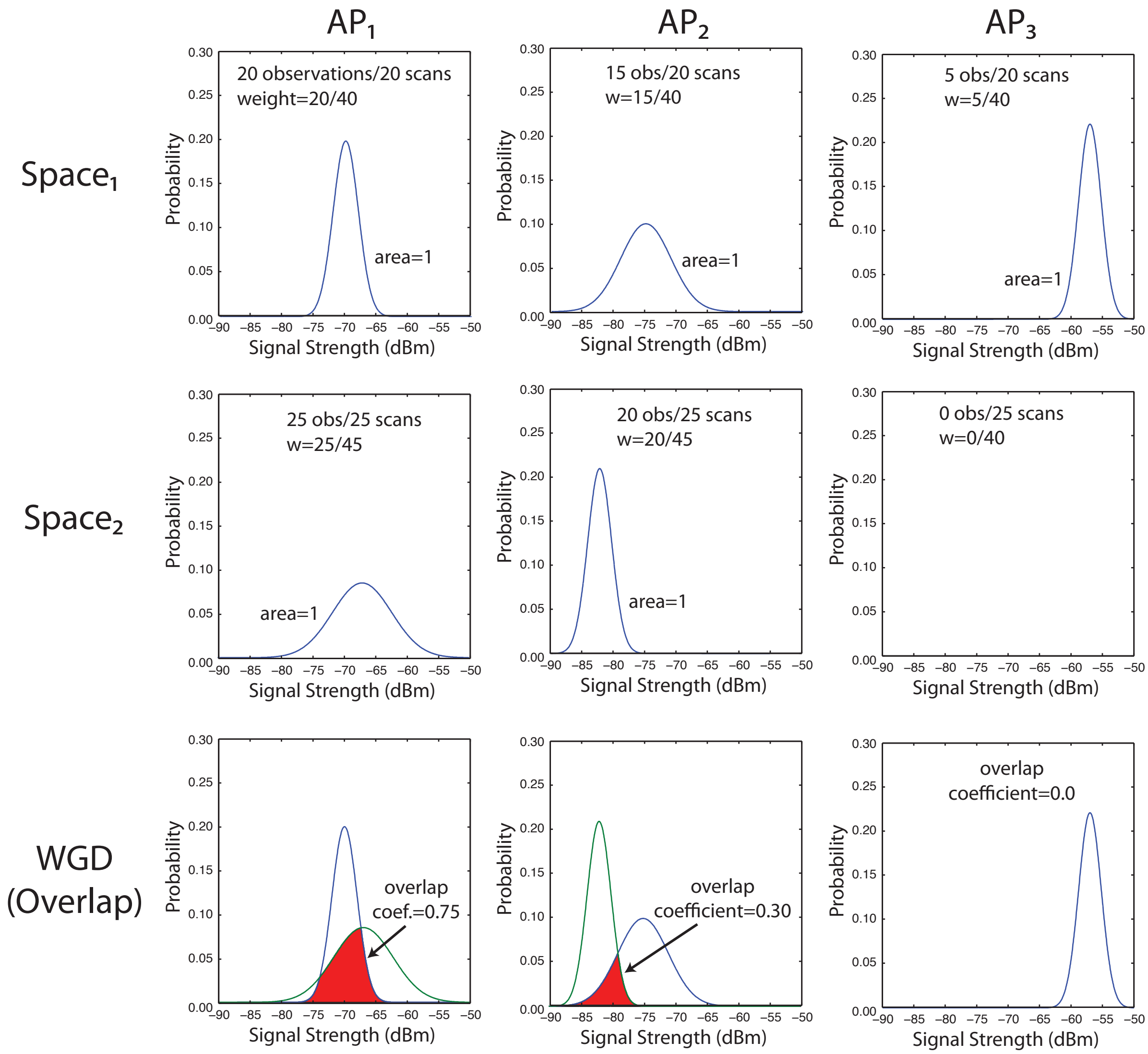

Figure 1: Example of WGD method. The 20 scans in $space_1$ have observed three different access points: AP_1, AP_2, AP_3. The 25 scans in $space_2$ have observed two different access points, two of which are the same as those seen in $space_1$: AP_1, AP_2. AP_3 was not observed in $space_2$. To compute the weights for $space_1$, we divide the observations for each AP by the total number of observations: $20 + 15 + 5 = 40$. The same procedure is done for $space_2$. This completes the creation of fingerprints for these two spaces. The bottom row shows how the distance between the two fingerprints for spaces 1 and 2 is computed:
$s = 0.75 \times \frac{20/40+25/45}{2} + 0.30 \times \frac{15/40+20/45}{2} - 5/40$.
"$Space_2$" could equivalently be a set of scans as seen by an end-user's device: the algorithm to compute their overlap score would be the same.

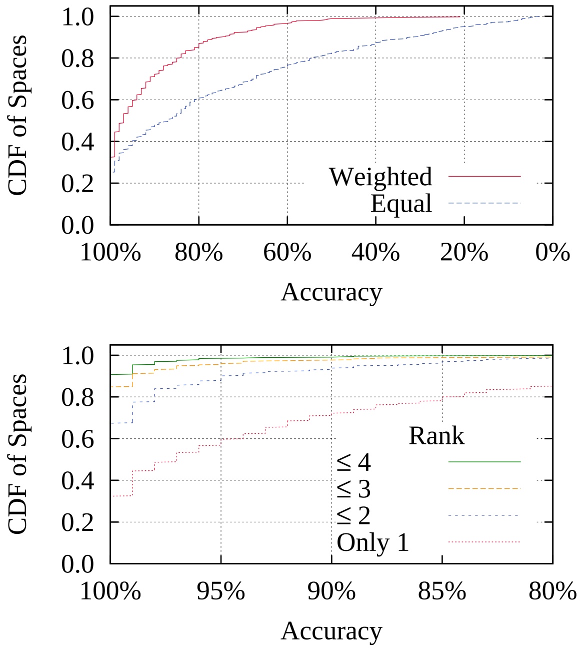

Figure 2: Simulation results from the WGD method using a large indoor dataset. **Top:** When the weighting functionality is switched off and all RF sources are counted equally, we found greatly reduced accuracy. For example, only 15% of spaces had an accuracy of less than 80% using the weighting function, but 40% had this same level of accuracy without it. **Bottom:** Many of the spaces in our dataset are topologically close to one another. Topologically close spaces often have close RF fingerprints, which occasionally confounds RF-based localization. If we allow for non-exact matches, the data show that the true correct space is often only a few steps away from our best guess.

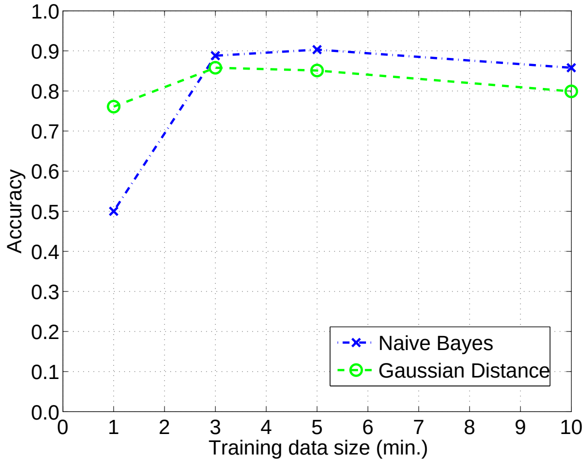

Figure 3: Preliminary results that compare Bayes to WGD show comparable accuracy subject to tuning. The results hint that WGD may perform better with less training data.

2.2 Related Methods

The closest related work to the WGD localization method is a paper by Lemelson *et al.* [11]. In this work, the authors use basic Gaussian overlap – without any weighting – to determine the similarity of the fingerprint of one point with adjacent points (in contrast to our per-space fingerprints, this research group prefers to determine fingerprints by overlaying an artificial grid over the space map).

Lemelson *et al.* do not use the overlap method for localization at all. Instead, the use it to determine the similarity of one point's Gaussian fingerprint with another. They are attempting to anticipate the likely estimate localization error for a given point. They show that points with very similar fingerprints (as determined by the overlap function) tend to have poor localization accuracy, because they are often confused with adjacent points.

As shown in Figure 2 (top), without the weighting, this method is clearly inferior to weighting based on the frequency of AP reception. An example where this method without the weighting would clearly perform poorly is a case where many spaces occasionally hear from many APs, and always hear from exactly one unique one. In this case, the no-weighting method would lose the single unique characteristic in the noise, whereas the weighting method would select the right space. Although this is an extreme example, it illustrates a problem that often occurs in environments with dense AP coverage.

3 Compressing Weighted Gaussian Distance Archives

A key problem in RF fingerprint-based positioning is the constant network connectivity required to keep the cache of fingerprints up-to-date. Recall that a fingerprint for a given space is a set of mappings from access points AP_i to data triples $\langle w_i, \mu_i, \sigma_i \rangle$ (See Eq. 1).

Note that if a system is using simple Bayesian positioning w_i can be dropped. This compression method also applies to Bayesian archives.

In urban environments that are dense with RF sources, the fingerprint for a single space can contain dozens or hundreds of these triples. Following from the weights of the Gaussian Distance method, we observe that not all of these RF sources supply the same amount of information. Much like Principle Component Analysis can highlight the (typically much smaller) number of dimensions required to represent a high-dimensional system, we observe that most RF sources do little to differentiate one space from others. Instead, these sources are extraneous and can be discarded during the fingerprint generation process. For illustrative purposes, we assume that all raw scans ($\langle RSSI, position \rangle$ pairs) are available at a central server and that the goal of the server's fingerprint generation process is to create set of signal map archives that can be downloaded to a client.

While other methods for selecting the top-k AP sources exist, using the highest-weighted ones is a simple method that works in practice. Thus, we can reduce a WGD fingerprint from dozens or hundred of RF sources to k (*e.g.,* 5). Since this list of sources constitutes the bulk of a signal map archive, this method reduces the size of an archive by approximately an order-of-magnitude.

A second observation that results in significant space savings is that the floating point numbers that make up the fingerprint can be represented by only a few (*e.g.,* 7) bits without degradation in accuracy (giving each 128 possibilities). For example, the range of the mean only needs to extend from 0 to -100 dBm (at most), or could be represented as a base (for example, 20 dBm) plus an offset at 0.5 intervals up to 84. The range of the standard deviation is typically < 10 dBm. Thus, even 0.1 increments or increments of expanding size (more detailed toward 0) would fit into seven bits. Assuming that each number was originally stored in 32 bits, this reduces the fingerprint length to at most one quarter of the original size. All three components of the WGD fingerprint are reducible. For particular physical sites, it may be feasible to reduce the size even further. This optimization reduces archive size by half an order-of-magnitude on average.

A signal map archive that uses this compression method would have at least three main sections: a header, a space description, and a fingerprint map. Note that the archive could also include bounding polygons for each space. Also note that a converter could serialize an XML version of the archive into the described format.

Field	Description
Name	Signal Map Title
Version	Version Number
GPS	Absolute coordinate; space offsets relative to this coordinate
# spaces	Total count of spaces in map

Table 1: Fingerprint Archive Header

Field Name	Univ. Space ID	Space ID	Canonical Name	Offset Coordinate (x,y,z)
Size (bits)	64	16	...	...

Table 2: Fingerprint Archive Space Descriptions. Map universal space IDs to IDs local to archive. Bounding area could also be included here.

Field Name	AP ID	# spaces	Space ID$_1$	Weight	Mean	Std. Dev.	Space ID$_2$	...
Size (bits)	48	7	16	7	7	7	16	...

Table 3: Fingerprint Archive Body. List spaces and per-space fingerprints associated with this access point.

Field Name	Space ID	AP ID$_1$	Weight	Mean	Std. Dev.	AP ID$_2$	...
Size (bits)	16	16	7	7	7	16	...

Table 4: Alternative Fingerprint Archive Body. Store compressed fingerprint directly after each space ID. Together with each space ID are *e.g.,* $k = 5 \times \langle AP_i, w_i, \mu_i, \sigma_i \rangle$. This alternative may result in improved on-device performance depending on the implementation. It also requires a mapping from the 2^{48} AP space to a unique ID, which could be sixteen bits or smaller depending on the number of APs in the archive (here it is shown to be sixteen bits).

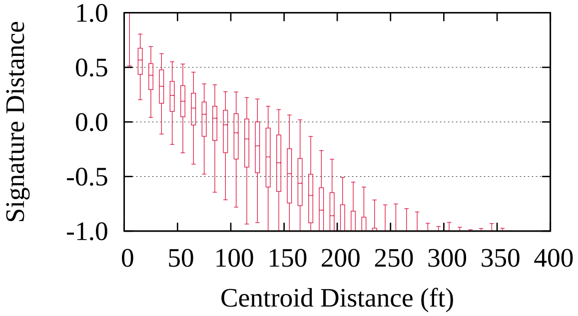

Figure 4: **Signal-to-Physical Distance Correlation.** The data show that there is a strong correlation between fingerprint distance and physical distance, particularly when spaces are nearby to one another. Space distance is measured as centroid-to-centroid using data from a nine-story building which contains more than 1,400 distinct spaces. Fingerprint distance is computed using WGD. The top and bottom whiskers show the top 5% and bottom 5%, respectively. The boxplot lines mark the 25^{th}, 50^{th}, and 75^{th} percentiles.

An alternative signal map archive is essentially a "sideways" turn on the first type of archive. This alternative takes more space, but may provide more efficient on device.

We are not aware of previous work whose aim was to reduce the size of the overall signal map.

4 Estimating Physical Distance with Weighted Gaussian Distance

Using WGD and other RF fingerprint-based localization methods suggests a simple method for estimating the physical distance between two (or more) devices. This method does not require a near field communication channel or short range proximity sensor such as Bluetooth.

Figure 4 shows the correlation between physical distance and fingerprint distance. The data suggest that WGD and other functions that compute a distance metric on RF scans or fingerprints can infer the physical distance between where those scans were taken, particularly when the physical distances are close.

One example use for this physical distance estimation is a device pairing system. The system could require two devices to be in the same room to exchange data could compare their RF fingerprints to estimate their physical distance. For example, in our data, spaces which had a fingerprint distance > 0.5 were always less than 100 feet apart. The system would only allow the pairing to occur if the RF fingerprint distance were below a certain threshold. In particular, this could be combined with a haptic action or device movement by the device holders that signaled to the system that two or more devices should temporarily be allowed to share data (*i.e.,* they could be shaken in the same direction or "bumped", as detected by an accelerometer). Determining that the two (or more) devices were approximately in the same physical location could take place at a server or on the devices themselves if they were connected by a communication channel. The pairing could occur with any device that can form an RF fingerprint (*e.g.,* mobile phone, laptop, stationary device with a wireless card). After the pairing has occurred, the devices can continue exchanging data when they are not in the same physical location, depending on the application.

This RF-level proximity detection has a few advantages over other methods, such as Bluetooth. First, an application can take advantage of the RF scanning that is already on-going: Bluetooth or a similar protocol does not need to be switched on or off. Second, it seems simpler to associate larger numbers of devices. Since the comparison is happening at the signal level and is essentially a bounded nearest neighbor search, extending from two paired devices to many is trivial. Lastly, computing the device proximity at a remote server could have security advantages. For example, faking the current set of RSSI values and APs seen may be difficult for an attacker.

A related, but distinct, body of work has focused on estimating the speed at which a person is moving based on the changes in their perceived RF fingerprint (for example, Afgani *et al.* [1]). The immediate implication of this body of preceding work is that there is a correlation between physical and RF distance. In contrast, our method here provides a distance function between two (or more) complete RF fingerprints from different devices, and then allows their physical distances to be estimated. In particular, a server that receives two user fingerprints at approximately the same time could determine that the two devices were in the same physical location and then allow other software to perform a function, such as symmetric key exchange, public key exchange, or data exchange. For example, a nearby device could be selected as the source for a peer-to-peer software download, data exchange, or cache, because it is known that the two devices are likely to be "close" in their network distance (latency) as well as physically.

References

[1] M. Z. Afgani, S. Sinanovic, K. Khashaba, and H. Haas. Radio frequency signature correlation based speed estimation for indoor positioning. *Journal of Communications*, 4(2):96–107, 2009.

[2] P. Bahl and V. N. Padmanabhan. RADAR: An In-Building RF-Based User Location and Tracking System. In *INFOCOM*, Tel Aviv, Israel, Mar. 2000.

[3] P. Bahl, V. N. Padmanabhan, and A. Balachandran. Enhancements to the RADAR User Location and Tracking System. Technical Report MSR-TR-00-12, Microsoft Research, February 2000.

[4] A. Barry, B. Fischer, et al. A long-duration study of user-trained 802.11 localization. In *MELT*, 2009.

[5] E. S. Bhasker, S. W. Brown, and W. G. Griswold. Employing user feedback for fast, accurate, low-maintenance geolocationing. In *PerCom*, 2004.

[6] P. Bolliger. RedPin: Adaptive, Zero-Configuration Indoor Localization. In *LoCA*, 2008.

[7] A. Haeberlen, E. Flannery, A. M. Ladd, A. Rudys, D. S. Wallach, and L. E. Kavraki. Practical Robust Localization over Large-Scale 802.11 Wireless Networks. In *MOBICOM*, 2004.

[8] J. Hightower, R. Want, and G. Borriello. SpotON: An Indoor 3D Location Sensing Technology Based on RF Signal Strength. Technical Report UW CSE 2000-02-02, University of Washington, February 2000.

[9] H. F. Inman and E. L. Bradley. The overlapping coefficient as a measure of agreement between probability distributions and point estimation of the overlap of two normal densities. *Communications in Statistics-Theory and Methods*, 18(10):3851–3874, 1989.

[10] A. LaMarca, Y. Chawathe, S. Consolvo, J. Hightower, I. E. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, J. Tabert, P. Powledge, G. Borriello, and B. N. Schilit. Place Lab: Device Positioning Using Radio Beacons in the Wild. In *Pervasive*, 2005.

[11] H. Lemelson, M. Kjaergaard, R. Hansen, and T. King. Error Estimation for Indoor 802.11 Location Fingerprinting. In *Proc. of the International Symposium on Location and Context Awareness*, Tokyo Japan, May 2009.

[12] J. M. Linacre. Overlapping Normal Distributions. *Rasch Measurement Transactions*, 10(1), 1996.

[13] S. Teller, J. Battat, B. Charrow, D. Curtis, R. Ryan, J. Ledlie, and J. Hicks. Organic Indoor Location Discovery. Tech. Report CSAIL TR-2008-075, Massachusetts Institute of Technology, Dec. 2008.