Opportunities and Challenges in Crowdsourced Wardriving

Piotr Sapiezynski
Technical University of Denmark

Radu Gatej
University of Copenhagen

Alan Mislove
Northeastern University

Sune Lehmann
Technical University of Denmark

ABSTRACT

Knowing the physical location of a mobile device is crucial for a number of context-aware applications. This information is usually obtained using the Global Positioning System (GPS), or by calculating the position based on proximity of WiFi access points with known location (where the position of the access points is stored in a database at a central server). To date, most of the research regarding the creation of such a database has investigated datasets collected both artificially and over short periods of time (e.g., during a one-day drive around a city). In contrast, most in-use databases are collected by mobile devices automatically, and are maintained by large mobile OS providers.

As a result, the research community has a poor understanding of the challenges in creating and using large-scale WiFi localization databases. We address this situation using the deployment of over 800 mobile devices to real users over a 1.5 year period. Each device periodically records WiFi scans and its GPS coordinates, reporting the collected data to us. We identify a number of challenges in using such data to build a WiFi localization database (e.g., mobility of access points), and introduce techniques to mitigate them. We also explore the level of coverage needed to accurately estimate a user's location, showing that only a small subset of the database is needed to achieve high accuracy.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

Keywords

wifi; wardriving; mobility; location

1. INTRODUCTION

Localization is an increasingly important trend on mobile devices today. Mobile applications use localization to provide users with accurate driving directions, recommendations for local points of interest (e.g., restaurants), and even as a form of authentication [10]. Determining a mobile device's location is typically accomplished in one of two ways: First, mobile devices can use various satellite-based systems (GPS, Galileo, or GLONASS). While most mobile devices today ship with dedicated GPS hardware, relying on GPS alone for determining location has a number of downsides: obtaining an initial GPS fix introduces non-negligible delay, and causes significant power consumption.

Second, mobile devices can use *WiFi localization*. In brief, WiFi localization works by having the mobile device listen for advertised WiFi networks (each WiFi access point periodically announces its unique identifier or *BSSID*, as well as the name of the network, referred to as *SSID*), and report that list to a central server. The server then computes the most likely location of the mobile device and returns the result. Thus, for WiFi localization to be effective, the server must have a pre-computed database of WiFi access points (APs) and their locations. Unfortunately, building such a database is time-consuming and expensive: the database must be comprehensive (covering many locations) and up-to-date (as new APs are deployed and existing ones move).

Originally, the aim of such databases was to enable indoor positioning through finger-printing [3, 9, 20] and later through RF-modeling [15, 5]. Most recent work on indoor localization achieves sub-meter accuracy by rotating the sensing device to simulate directional antennas [14]. As the APs became more wide spread it became possible to use them for outdoor localization as well. The databases were then created by manually going to different locations and recording the observed APs (often termed *wardriving*) [4, 18, 7, 11]. Today, however, these databases are often built by having dedicated software on the mobile devices collect and report data back both in indoor [21, 27] and outdoor [2, 19, 26] contexts. Therefore, creating such a database at scale is typically only the domain of mobile OS providers (e.g., Apple, Google) or dedicated companies (e.g., Skyhook Wireless).

As a result, the research community currently has a relatively poor understanding of large-scale WiFi localization databases. In this paper, we address this situation by providing insights into the challenges underlying the creation of such a database, and the trade-offs in using them. We first collect a data set based on a deployment of over 800 mobile phones to students at a university in Copenhagen, Denmark for over 1.5 years. These phones run a stock Android OS with custom collection software instrumented to gather GPS location and overheard WiFi APs.

Overall, we collect over 1.8M simultaneous measurements of WiFi APs and GPS location, and observe more than 1.3M

IMC'15, October 28–30, 2015, Tokyo, Japan.

ACM 978-1-4503-3848-6/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2815675.2815711.

unique WiFi APs. Many of the APs are only seen a small number of times, so we focus on the 376K APs that we observe at least five times. To the best of our knowledge, this represents the most comprehensive data set of this kind that has been examined in the research literature. Using this data set, we build a WiFi localization database for Copenhagen. We discuss and identify a number of key challenges and issues in doing so:

The scale of the dataset. Most existing studies were performed either in controlled environments or over a short time. Here, we show that the WiFi landscape is constantly changing, new access points are added and old ones are moved to new locations or retired.

Mobility. With increasing trend of mobile WiFi APs, such as MiFi devices, routers on buses and trains, and mobile phones which also serve as hotspots, we observe that discovering and filtering mobile APs presents a significant challenge. Failing to properly filter these can lead to gross errors when estimating a device's location.

Noisy data. Unsurprisingly, relying on commodity hardware introduces noise into the measurements of location, signal strength, and detectability of APs, which must be handled when inferring the location and mobility of APs.

We also explore using the database we build to estimate the locations of devices given a set of overheard APs. Specifically, we examine the trade off between the number of APs in the database and the estimation accuracy. We show that knowing the location of only a small fraction of all the APs (3.7%) is actually needed to locate users to within 15 meters 75% of time.

2. METHODS

We now describe the data we use to build our WiFi localization database.

Phone deployment. We use data collected by the Copenhagen Networks Study experiment [23]. In this experiment, students opt-in to receive a smartphone in exchange for agreeing to let us use to collect data (e.g., Bluetooth and WiFi scan results, location estimations, call and SMS metadata, etc). The students agree to use the device as their primary phone. The experiment has been reviewed and approved by the Danish Data Protection Agency, and participants are provided with a web interface where they can access and remove any of their collected data.

The data analyzed in this work covers a period from September 2013 through March 2015 and involves more than 800 students, with 300–600 participants active on any given day. Because of software failures and physical destruction some phones had to be replaced, and thus 1,000 devices were used in total. The primary focus of the Copenhagen Networks Study experiment is the study of human interactions, hence the setup was not explicitly optimized towards discovering the locations of APs. Nevertheless, we show in this paper that the WiFi scans and GPS data allow us to do so.

Data collection app. On the phones, we install an app based on the Funf framework [1]. It starts automatically when the phones boot, so the users do not need to take action to begin collecting and uploading data.

The app collects data both actively (it requests location and WiFi updates every 5 minutes) and opportunistically (whenever another app requests updates). In order to save the battery, most of the location data is obtained using the network and/or fused provider (i.e., an existing WiFi localization database). Since we intend to use the GPS measurements as ground truth, we focus only on the 10.5% of location readings that are provided by the GPS hardware.

As a consequence, while the median sampling period between GPS readings is 1 second, only 29% of per-user hourly bins have at least one GPS sample (i.e., we only know the GPS location of users in 29% of the hours, on average). This distribution is a consequence of apps like Google Maps that either use GPS data constantly or not at all.

Since we are studying WiFi localization databases, in the remainder of the paper we focus on the 1,794,473 GPS samples which happened within the same second[1] as a WiFi access point scan. According to our measurements, a single WiFi scan lasts approximately 500 ms and this time does not depend on the number of saved networks.

It is important to note that the securing the wireless network does not make it impossible to scan it: regardless of the encryption, each router broadcasts its unique identifier and the name of the network in clear text.[2]

Filtering data. In the 567 days of observations, our participants observed 7,203,471 unique APs, out of which 1,320,838 (18.3%) were scanned at least once in the same second as a GPS estimation. However, the majority of these APs were observed with a GPS estimation a very small number of times: 944,904 (71.5%) have less than five observations. Thus, in the remainder of the paper, we focus only on the 375,934 APs that were observed at least five times *together with* a GPS estimate in the same second to build our WiFi localization database.

3. BUILDING THE DATABASE

We now examine the collected data, with the goal of building a WiFi localization database.

3.1 Estimating the locations of APs

The primary challenge we face is estimating the positions of the APs, given our WiFi scan data. Intuitively, this seems straightforward, but AP mobility presents a number of challenges. In general, we expect APs to fall into one of three categories:

- **Static.** We expect that many APs are static and have a fixed location that does not change over the course of the experiment.
- **Moved.** Given that our data covers 1.5 years, some APs may remain static for long periods of time, but may be moved a small number times. For example, businesses may redeploy APs, and residents of Copenhagen may change apartments, taking their APs with them.

[1] Allowing for even a short time difference would introduce noise into the measurements. For example, a car driving within city speed limits moves at 14 m/s. Because of uneven and sparse sampling, it is not feasible to calculate the speed of the measuring device and discard the scans that were performed by phones in motion.

[2] We note that is possible to *hide* a network by disabling the access point's SSID broadcasts (though this provides little actual security [17]). Routers configured this way still broadcast their BSSID and are present in our dataset.

- **Mobile.** We also expect to see some APs that show no static behavior; these could include APs located on buses and trains, as well as MiFi devices and mobile phone hotspots.

We categorize APs into these three classes by *clustering* the observed WiFi scan data. Specifically, every time a GPS estimation happens in the same second as a WiFi scan, we add the latitude/longitude to the list of observations of each AP visible in the scan. We then categorize the APs as follows:

Static access points. We first compute the geometric median [16] of all locations associated with each AP; if "most" of the observations are "close" together, we then declare the AP to be static, and declare the geometric median to be the AP's location.[3]

However, selecting the right thresholds for "most" and "close" to use is more complex than it may seem, as it is difficult to determine the operating range of an AP. First, devices compliant with popular standards can be expected to have a range from 20 meters indoors (the 802.11 standard) to 250 meters outdoors (802.11n) [25]. We therefore set the radius for a static AP to be no more than 300 meters. Second, due to the complex nature of signal propagation, the range can be shortened or enlarged due to characteristics of the local environment (e.g. buildings, narrow corridors). Third, GPS devices are known to sometimes return erroneous readings [6]; to deal with these, we allow for up to 5% of locations associated with an AP to be in a bigger distance than 300 meters from the median position.

We classify the APs that satisfy this condition (95% of readings within 300 meters) as **static**, and find that 263,281 (70%) of the APs fall into this category.

Moved and mobile access points. We assume that the rest of the APs are either moved or mobile. To disambiguate the two cases, we repeat the clustering above but allow for *multiple* such clusters.

Specifically, we group any two locations within 600 meters (twice the radius) into the same cluster, and discard any clusters that have fewer than 5 measurements. If at least 95% of the points can be associated with one of the clusters, and the clusters can be cleanly separated in time, we categorize the AP is **moved**. We observe that 1,087 (0.3%) APs fall into this category. Otherwise, we categorize the AP as **mobile**. We observe that 111,566 (29.7%) APs fall into this category.

3.2 Classification evaluation

We now briefly evaluate our classification. As a sanity check, in Figure 1 we show the locations of all APs with the SSID of `dtu`, which is the SSID of APs installed at our university. The left panel shows the APs on a metro area scale; each group of APs is correctly placed at one of the university campuses and out-of-campus buildings. The right panel shows the APs around the main campus of the university. While this is not a definite confirmation of the accuracy of our approach, this example of 1,100 APs shows that we should not expect too many gross errors.

We evaluate our method of identifying the mobile APs by verifying the classification of APs that are nearly certainly

[3]Following the definition of accuracy from the Android Location API, we calculate the radius around the median within which 68% of points are enclosed [8].

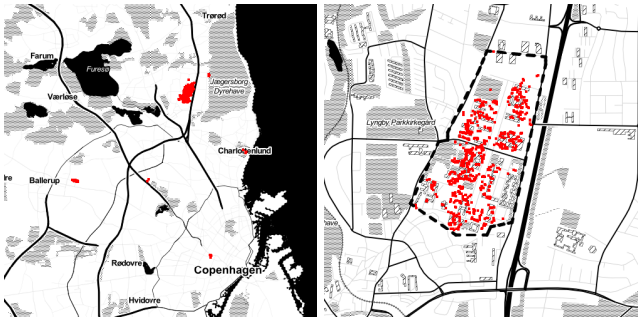

Figure 1: Sanity check of the method: estimated locations of APs belonging to Technical University of Denmark On the metro area scale (left panel), different campuses and out-of-campus buildings are visible, while none of the APs is estimated to be at a location not associated with the university. A detailed view of the main campus (right panel) reveals that the APs are grouped within perimeter.

static and those that are nearly certainly mobile. First, we choose APs with `eduroam` SSID as examples of APs which we expect to be stationary, since these are the names of APs at universities. Out of 3,654 such APs with at least 5 observations, 3,117 (85.3%) were identified as static and 9 (0.2%) as moved. Universities are known to relocate APs, which may partially explain why our accuracy is not 100%. Next, we choose APs with `Bedrebustur` or `Commutenet` SSIDs as examples of APs we expect to be mobile, since these are the official names of networks on buses and trains in Copenhagen. Out of 650 such APs with at least 5 observations, 642 (98.8%) were identified as mobile, and 8 (1.2%) as static.

It is important to note that access points with more observations are less likely to be classified as mobile (e.g., 29.7% of access points seen at least 5 times are classified as mobile, while only 10.0% of access points seen at least 200 times are classified as mobile). This effect is likely due to the the biased sampling of access points by users (i.e., static access points are more likely to be sampled many times, due to their static nature).

Overall, our results suggest that our AP classification methodology is likely to have high accuracy.

3.3 Accuracy of database

Next, we explore two aspects of the accuracy of the WiFi localization database: (1) how the number of measurements of a given AP affect our estimate of its location, and (2) how the number of measurements of a given AP affects our ability to classify it as mobile or fixed location.

Number of measurements needed. While we cannot measure the error of location estimation without knowing the ground truth location, we can analyze how the location estimation changes with the number of observations. We select 46,000 APs classified as static and with more than 50 measurements. For each of these APs we select N random observations, calculate the distance between the location of the AP estimated from all the observations and the estimation based on N random observations. We vary N from 1 to 50 and repeat the process 10 times.

In Figure 2 we show that even in case of APs with fixed location, using too few measurements leads to significant deviations in the estimated position. For example, calculating the position of the AP based on only two observations leads

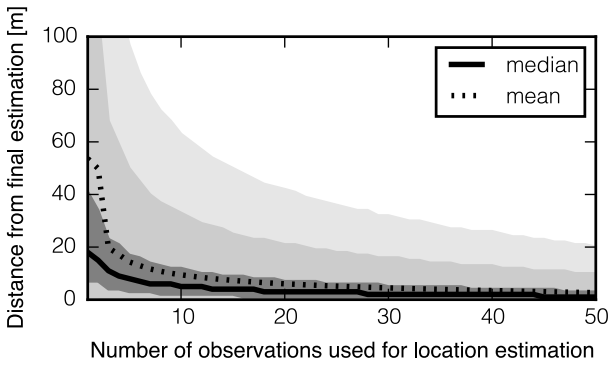

Figure 2: Too few observations lead to estimation errors. We randomly subsample the measurements of 40,000 static APs to measure the error caused by fewer measurements. The shaded bands represent percentiles 1-99, 5-95, and 25-75. To ensure estimation error below 50 meters in 99% cases, 15 observations are necessary. Five observations, which we use as minimal threshold, are enough to estimate the location of an AP with error not higher than 50 meters in 95% of cases.

to a 50 meter error, on average. 15 observations are necessary to ensure that the error is not larger than 50 meters in 99% of cases.

Mobility and sample size. Because of the prevalence of APs that are mobile, too few observations might lead to their incorrect classification as stationary. To evaluate this, we select 20,000 APs classified as mobile and more than 50 observations. For each of them we select N random observations and re-run our classification procedure. We vary N from 2 to 50 and repeat the process 10 times.

Because we only allow 5% of observations to be outside of the 300 meter radius around the median, with too few observations we might classify a fixed AP as mobile. We repeat the described experiment but with fixed APs and calculate the fraction of misclassified fixed location APs as a function of N.

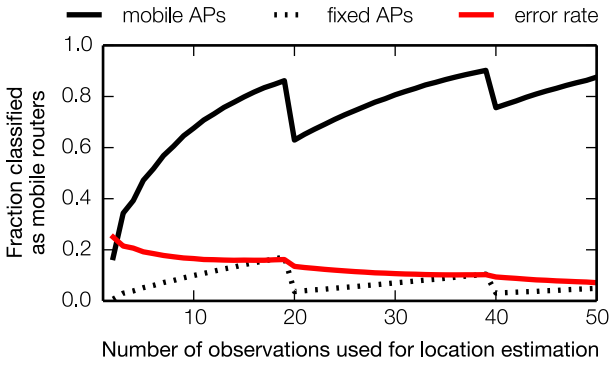

Figure 3: Too few observations lead to misclassifications between mobile and static APs. We randomly subsample the observations of 46,000 static APs and 20,000 mobile APs to measure the classification error caused by too few observations. With just 5 observations, 46% of mobile APs are classified as static and of 5% of static APs are misclassified as mobile. Given the class imbalance, that results in 18% misclassification rate.

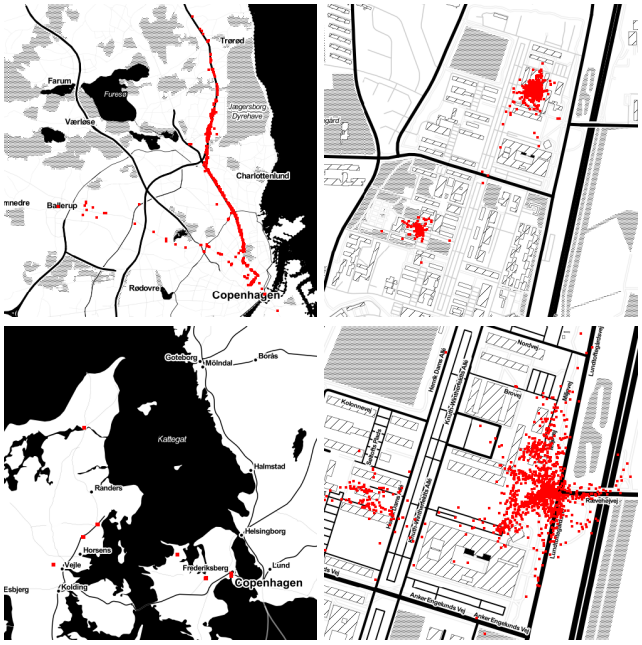

Figure 4: Longitudinal observations reveals mobile and moved APs. Shown are the observed locations of a mobile AP installed on a bus (top left), an moved AP (top right), an AP moved 6 times (bottom left), and an AP with ambiguous behavior (bottom right).

As we show in Figure 3, the more observations we base our estimations on, the more accurate the results are. The "spikes" at 20 and 40 APs are caused by the fact that the 5% noise threshold translates to 0 noisy samples with less than 20 observations, 1 noisy sample with 20-39 observations, etc.

Taken together, these results suggest that building an accurate WiFi localization database requires large amounts of data collected continuously over time. To better visualize the importance of longitudinal observation, we provide several examples of APs with different patterns of observation. The top left panel of Figure 4 shows a clear example of a mobile AP; in this case, it is installed in a bus. In such cases, a few observations should be sufficient to correctly classify the AP as mobile. In other cases however—as shown in top right and bottom left panels of Figure 4—a long observation period is beneficial. While in the top right example not knowing the new location of the AP would lead to errors at the range of hundreds of meters, the bottom left example shows an AP whose location changes hundreds of kilometers during the observation period. Still in some cases, even a long observation period might not be enough to determine the nature of the AP, as shown in the bottom right panel: the AP seems to have two major placements, but they overlap in time, so we classify this AP as mobile.

4. USING THE DATABASE

With our WiFi localization AP database built, we now turn to using the database to estimate the location of a user. In brief, when a user requests their location to be calculated, they present the database with (a) a list of the AP SSIDs and BSSIDs that it current observes, and (b) the received signal strength (RSSI) of each of these APs. We first explore

Figure 5: Using a population of students from one university results in uneven sampling. Each red point on the maps represents a single AP. The inferred locations of APs in the city center indicate that sampling is not uniform across space: the routers seem to be located along the streets, not inside the buildings.

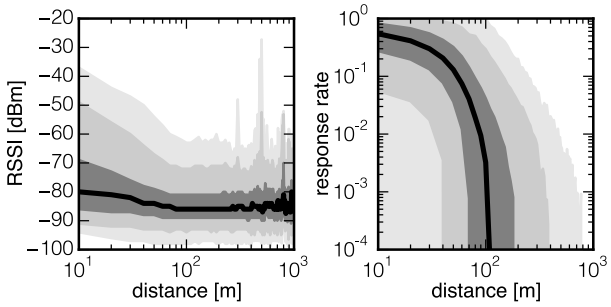

Figure 6: RSSI (left) and response rate (right) as functions of distance from the AP. The shaded bands represent percentiles 1-99, 5-95, and 25-75, the bold line represents the median value. There is a weak correlation between RSSI and distance with Spearman's correlation of $\rho = -0.23$ for distances from 0 to 100 meters, and no correlation for larger distances. There is a strong correlation between response rate and distance ($\rho = -0.64$) for distances from 0 to 100 meters, and a weaker ($\rho = -0.30$) correlation for larger distances. Using non-specialized hardware raises a number of challenges, including noisy measurements of RSSI and location. As a result, RSSI is not a reliable proxy for distance.

how the signal strength relates to the distance to AP before examining our ability to estimate the user's location.

4.1 Estimating distance from APs

RSSI. As radio waves propagate through space they become attenuated; the amount of attenuation can be used to calculate the distance d. There are a number of models describing the attenuation of WiFi signals and one of the simplest is the log-distance path loss (LDPL) model [12], from which the distance can be calculated using Equation 1:

$$d_{ij} = 10^{(\frac{P_i - p_{ij}}{10\gamma_i})} \quad (1)$$

In Equation 1 mobile user j is at distance d_{ij} (m) from access point i and sees the signal strength of p_{ij} (dBm). P_i is the power transmitted by the AP. The path loss exponent γ_i captures the rate of fall of RSSI around the AP i which depends on the environment the router is in [13]. If the transmitted power and path loss exponent are known, three non-collinear measurements of the AP should theoretically be enough to determine its position using trilateration.

However, accurately estimating the distance given RSSI has been shown to be a challenging problem. First, because the transmitted power and the propagation loss exponent are different for every router and need to be calculated, two more measurements are necessary to solve the system of LDPL equations. Second, since the receiver characteristics vary greatly even among devices of the same make and model [24, 9, 5], more measurements are necessary to compensate for individual characteristics [5]. Third, due to the inherent noise in the measurements and a dynamically changing environment (e.g., people walking by) the RSSI reading can be very noisy in practice. For example, our previous work observed that the RSSI reading can deviate as much as 10 dB from the mean even when the source and destination are static [22]. We note that while there are methods that take advantage of the variable attenuation introduced by a human body [28], they require accelerometer data to be collected as well (which we were unable to collect in our experiment).

Nevertheless, RSSI has been reported in other studies of war-driving as a useful, if somewhat noisy, proxy for distance [4]. To verify this finding, we randomly select 5.6M observations of 30,000 APs classified as static and present RSSI as function of distance from the inferred location in the left panel of Figure 6. There is only a weak correlation between the measured signal strength ($\rho = -0.23$) and the inferred location, and that correlation disappears for distances larger than 100 meters. The figure also reveals that a strong RSSI can be used as an indicator of close distance, but a weak RSSI does not indicate that the APs is far away. We use Spearman's rank correlation coefficient, instead of Pearson's product-moment correlation because we cannot expect a linear relationships between RSSI and distance. Pearson's ρ values are lower in the analyzed relationships.

The low correlation could still be caused by the differences between routers (the emitted power and the influence of obstacles). We therefore calculate the correlation between distance and RSSI for each router separately. We find that about 35% of the routers with at least 50 observations have statistically significant, negative relation between distance and RSSI with mean $\rho = -0.36$. On the other hand, 16% of such routers have a positive relation between the RSSI and distance, with mean $\rho = 0.32$. All reported correlations are statistically significant with $p_{val} < 0.01$.

Response rate. Here, we reevaluate the response rate as a proxy of distance from the AP, first suggested in [4]. Response rate at distance d is defined as the fraction of WiFi scans at distance d from the position of the AP which report finding the AP. We select a random subsample of 11,700 static APs with at least 50 observations. Then, for each AP, we find all scans recorded at distance d from its inferred location, varying the d from 0 to 1,000 meters. We define the response rate of a AP at distance d as a fraction of scans in which the AP was found. In the right panel of Figure 6 we show the correlation of distance and response rate. As expected, the response rate drops as the distance from the inferred location increases, with a much stronger correlation than RSSI ($\rho = -0.55$ for distances up to 100 meters). However, to measure the response rate, one must perform multiple scans at the same distance from the router.

4.2 Estimating the location of users

We now turn to estimating the locations of users using our database of the location of APs. Unfortunately, while response rate provides a much better correlation with AP distance, it is not ideal for estimating users locations: when estimating locations, doing so quickly is of paramount importance, and estimating response rate requires a number of scans. Thus, in the approach below, we simply use RSSI, and leave leveraging response rates to future work.

Knowing the list of APs recently observed, along with their RSSI, we explore estimating the user's location using four different approaches:

Mean coordinates. We ignore the RSSI and calculate the mean latitude and mean longitude among all the APs for which we know the location.

Geometric median. We ignore the RSSI and calculate the geometric median of the APs for which we know the location.

Mean weighted by RSSI. Each AP is assigned a weight based on the RSSI, with the weight defined by RSSI+100.[4]

We examine instances where different numbers of APs are observed in the scans, selecting 100 random instances between 0 and 30 observed APs. In the left panel of Figure 7, we show the cumulative error distributions for estimating the user's location using these three methods. The approach with geometric median location works best, followed closely by mean weighted by RSSI. While there are some differences in the performance of the three selected methods, they are negligible: all methods locate more than 50% of scans within 13 meters from the ground truth, 90% of scans within 70 meters, and 95% of scans within 120 meters.

In the right panel of Figure 7 we compare our best method (based on the geometric median) to the estimations which we acquired from the Google Geolocation API. We show the median error as a function of the number of APs used for the estimation. While our approach performs slightly better than Google's location API, the performance is similar. Google's crowd sourced data is collected using a wide variety of uncalibrated hardware (all of our phones are exactly the same model), which might lead to more measurement noise for Google's database. Since the number of APs in each scan is highly correlated with the population density [22], and the estimation errors are lower with more routers available, we expect that the location estimations will be best in densly populated areas.

4.3 Applicability of the localization database

In total, we identified 263,281 APs as static, constituting only 3.7% of the total of 7.2M unique APs observed. We revisit the original dataset with all the scans collected to verify whether this small set of APs can be used for localization in the broader context. We randomly select 51M of those scans and find that at least two of our static APs are visible in 73% of all scans, meaning we would provide an average error of 15 meters for 73% of all WiFi scans we observed.

The median error of 15 meters means that certain problems—such as car navigation—cannot be solved using WiFi signals alone. There are, however, a number of applications where the advantages outweigh the problems related to a relatively high positioning error. First, using geolocalized WiFi routers enables tracking the location of mobile devices with sub-minute time resolution at low costs in terms of battery or data consumption. As a consequence, it becomes possible to accurately measure for example time spent at each location, or detect whether the user changed their location in between two location scans pointing to the same place. Second, we show it is feasible to store a lookup database on the mobile devices themselves, thus enabling positioning without access to the Internet. Our database for the Greater Copenhagen area is only 9 MB, it could be a part of a mobile application targeted at tourists.

[4]The range of RSSI given by Android is -99dBm to 0dBm.

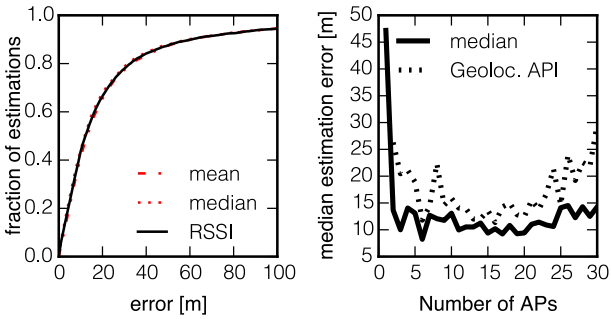

Figure 7: Location estimation accuracy does not strongly depend on the method. All methods perform similarly, and are able to locate 50% of scans with error no larger than 13 meters. Additionally, we compare our estimates to the results from the Google Geolocation API in the right panel; while our approach performs better, the differences are small.

5. SUMMARY

Being able to quickly and efficiently determine the location of a mobile device is becoming increasingly important. While mobile devices often contain dedicated GPS hardware to do so, they often opt to instead rely on WiFi localization databases as they are much quicker and more power-efficient. However, building such a database requires access to large-scale WiFi scan data over time, and is typically only available to the large mobile OS vendors.

In this work, we explored the opportunities and challenges in building such a database using a deployment of over 800 mobile devices. We found that mobility of *access points* was a key challenge in ensuring that the database is accurate; a significant fraction (30%) of APs are actually non-static. However, we found that using just the APs that we are confident are static, we can provide a location estimate for 73% of *all* scans with a median accuracy of 15 meters. Overall, our results provide the largest-scale look at WiFi localization databases that we know of in the research community.

Acknowledgements

We thank the anonymous reviewers for their helpful comments. This research was supported in part by NSF grants CNS-1054233, CNS-1319019, and CNS-1421444. All map tiles used in the article are by Stamen under CC BY 3.0. Map data by OpenStreetMap under CC BY SA.

6. REFERENCES

[1] N. Aharony, W. Pan, C. Ip, I. Khayal, and A. Pentland. Social fMRI: Investigating and shaping social mechanisms in the real world. *Pervasive and Mobile Computing*, 7(6):643–659, Dec. 2011.

[2] Apple. Apple Q&A on Location Data. http://goo.gl/dkPCAZ.

[3] P. Bahl and V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 775–784. Ieee, 2000.

[4] Y.-C. Cheng, Y. Chawathe, A. LaMarca, and J. Krumm. Accuracy Characterization for Metropolitan-scale Wi-Fi Localization. In *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*, MobiSys '05, pages 233–245, New York, NY, USA, 2005. ACM.

[5] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan. Indoor localization without the pain. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, pages 173–184. ACM, 2010.

[6] S. Duncan, T. I. Stewart, M. Oliver, S. Mavoa, D. MacRae, H. M. Badland, and M. J. Duncan. Portable global positioning system receivers: static validity and environmental conditions. *American journal of preventive medicine*, 44(2):e19–e29, 2013.

[7] A. Eustace. WiFi data collection: An update. http://goo.gl/VFJ9mM.

[8] Google. Android Developer Reference: Location. http://goo.gl/21VB7P.

[9] 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 *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking*, MobiCom '04, pages 70–84, New York, NY, USA, 2004. ACM.

[10] A. Hammad and P. Faith. Location based authentication, Oct. 24 2008. US Patent App. 12/258,322.

[11] D. Han, D. G. Andersen, M. Kaminsky, K. Papagiannaki, and S. Seshan. Access point localization using local signal strength gradient. In *Passive and Active Network Measurement*, pages 99–108. Springer, 2009.

[12] M. Hata. Empirical formula for propagation loss in land mobile radio services. *Vehicular Technology, IEEE Transactions on*, 29(3):317–325, 1980.

[13] M. Hidayab, A. H. Ali, and K. B. A. Azmi. Wifi signal propagation at 2.4 GHz. In *Microwave Conference, 2009. APMC 2009. Asia Pacific*, pages 528–531. IEEE, 2009.

[14] S. Kumar, S. Gil, D. Katabi, and D. Rus. Accurate indoor localization with zero start-up cost. In *Proceedings of the 20th annual international conference on Mobile computing and networking*, pages 483–494. ACM, 2014.

[15] H. Lim, L.-C. Kung, J. C. Hou, and H. Luo. Zero-configuration, robust indoor localization: Theory and experimentation. In *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM'06)*, 2006.

[16] J.-H. Lin and J. S. Vitter. Approximation algorithms for geometric median problems. *Information Processing Letters*, 44(5):245–249, 1992.

[17] J. Lindqvist, T. Aura, G. Danezis, T. Koponen, A. Myllyniemi, J. Mäki, and M. Roe. Privacy-preserving 802.11 access-point discovery. In *Proceedings of the second ACM conference on Wireless Network Security (WiSec'09)*, 2009.

[18] B. Meyerson. Aol introduces location plug-in for instant messaging so users can see where buddies are. http://goo.gl/2W1uYh.

[19] Microsoft. Location and my privacy faq. http://goo.gl/vvaZwZ.

[20] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 32–43. ACM, 2000.

[21] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen. Zee: Zero-effort crowdsourcing for indoor localization. In *Proceedings of the 18th annual international conference on Mobile computing and networking*, pages 293–304. ACM, 2012.

[22] P. Sapiezynski, A. Stopczynski, R. Gatej, and S. Lehmann. Tracking Human Mobility Using WiFi Signals. *PLoS ONE*, 10(7), 07 2015.

[23] A. Stopczynski, V. Sekara, P. Sapiezynski, A. Cuttone, M. M. Madsen, J. E. Larsen, and S. Lehmann. Measuring large-scale social networks with high resolution. *PLoS ONE*, 9(4), 04 2014.

[24] P. Tao, A. Rudys, A. M. Ladd, and D. S. Wallach. Wireless LAN Location-sensing for Security Applications. In *Proceedings of the 2nd ACM Workshop on Wireless Security*, WiSe '03, pages 11–20, New York, NY, USA, 2003. ACM.

[25] Wikipedia. Ieee 802.11. http://goo.gl/molLPd.

[26] D. Wu, Q. Liu, Y. Zhang, J. McCann, A. Regan, and N. Venkatasubramanian. CrowdWiFi: efficient crowdsensing of roadside WiFi networks. In *Proceedings of the 15th International Middleware Conference*, pages 229–240. ACM, 2014.

[27] S. Yang, P. Dessai, M. Verma, and M. Gerla. Freeloc: Calibration-free crowdsourced indoor localization. In *INFOCOM, 2013 Proceedings IEEE*, pages 2481–2489. IEEE, 2013.

[28] Z. Zhang, X. Zhou, W. Zhang, Y. Zhang, G. Wang, B. Y. Zhao, and H. Zheng. I am the antenna: Accurate outdoor AP location using smartphones. In *Proceedings of the 17th annual international conference on Mobile computing and networking*, pages 109–120. ACM, 2011.