

# An Open Platform for Studying and Testing Context-Aware Indoor Positioning Algorithms

Nearchos Paspallis, Marios Raspopoulos

**Abstract.** This paper presents an open platform for studying and analyzing indoor positioning algorithms. While other such platforms exist, our proposal features novelties related to the collection and use of additional context data. The platform is realized in the form of a mobile client, currently implemented on Android. It enables manual collection of radio-maps—i.e. fingerprints of Wi-Fi signals—while also allowing for amending the fingerprints with various context data which could help improve the accuracy of positioning algorithms. While this is a research-in-progress platform, an initial experiment was carried out and its results were used to justify its applicability and relevance.

**Keywords:** Indoor positioning · Fingerprint · Context-aware · Android

## 1 Introduction

The latest advances in Wireless Communication Systems and in Information Technology gave rise to various applications which require accurate information about the location of the connected devices. Especially in the context of mobile computing and the Internet of Things (IoT) in areas where satellite-based systems fail to provide accurate localization, indoor positioning is regarded as a key enabling technology [1].

The indoor positioning methods and algorithms proposed and developed either experimentally or commercially over the last decades, make use of various kinds of location-dependent radio context such as the Received Signal Strength (RSS), the Time of Arrival (ToA), the Time Difference of Arrival (TDoA), the Angle of Arrival (AoA) etc. [2] [3]. Fingerprint-based positioning [4] has become a very popular topic of research

---

A prior version of this paper has been published in the ISD2016 Proceedings (<http://aisel.aisnet.org/isd2014/proceedings2016>).

---

N. Paspallis (✉)  
University of Central Lancashire  
e-mail: [npaspallis@uclan.ac.uk](mailto:npaspallis@uclan.ac.uk)

M. Raspopoulos  
University of Central Lancashire  
e-mail: [mraspopoulos@uclan.ac.uk](mailto:mraspopoulos@uclan.ac.uk)

© Springer International Publishing Switzerland 2017  
J. Gołuchowski et al. (eds.), Complexity in Information Systems Development  
Lecture Notes in Information Systems and Organisation, XX  
DOI XX.XXXX/XXXXXXXX

in indoor positioning. It consists of two main phases: the *offline* phase where pre-measured location-dependent information (e.g. RSS), known as fingerprints that cover the entire area of interest, are stored in the database (radio-map), and the *online* phase where the instantaneous measurement is correlated with the fingerprints in the radio-map to estimate the position. These offline and online phases are reminiscent of the training and application phases commonly found in machine learning algorithms.

Fingerprint-based positioning using RSS can be classified into two main categories: deterministic and probabilistic. Deterministic methods estimate the location as a convex combination of the reference locations [5]. A very popular technique is the *K*-Nearest Neighbor (*KNN*) algorithm which averages the locations of the *K* fingerprints in the radio-map that better match the received measurement. In the probabilistic approach the location can be estimated by calculating and maximizing the conditional posterior probabilities given an observed fingerprint and a radio-map. This is usually a Bayesian Inference problem in which *a priori* knowledge can be introduced by defining different probabilities to different locations in the environment. Indoor positioning could be device-based in which the device collects the necessary information in order to perform the position estimation on its own or infrastructure-based where the context is pushed to the infrastructure (e.g. a centralized server) which performs the positioning.

In all the above techniques, the sole utilization of the radio parameters, during the position estimation, imposes limits that are hard to overcome. These limitations are often related to the inability of off-the-shelf mobile devices (which are usually based on the IEEE 802.11 standard) to accurately measure these parameters. To this end, the current research trend in indoor positioning is to put forward solutions which enable data fusion of radio-context with non-radio context such as information from inertial sensors and/or any other built-in sensors on the mobile or even context which manually added as prior knowledge (e.g. environment maps) [6].

Various advanced positioning algorithms have been proposed in the literature. These combine various types of information to provide more accurate results. For example, *Inertial Navigation Systems* (INSs) combining different kinds of sensors (e.g. accelerometers, magnetometers, and gyroscopes) have proven to adequately complement existing navigation means such as GPS/GNSS and the same concept can be applied to indoor positioning. For instance, the authors of [7] combine a RSS fingerprinting positioning algorithm with a Kalman filter-based tracking algorithm which estimates the location based on the information collected from inertial sensors.

Generally, the more information that is being considered into a position estimation problem, the higher the probability of a more accurate result. Additionally, the performance evaluation of wireless positioning systems is considered as a challenge by the research community, due to the diversity of positioning algorithms and the complexity of the factors that affect their performance. For this reason, in this paper we propose an open platform which enables easy collection and sharing amongst the research community of context useful for positioning algorithm testing and experimentation.

The rest of this article is organized as follows: Section 2 follows up with a description of related work and then Section 3 presents the design and implementation of the mobile platform. An evaluation via a custom-tailored experiment is presented in Section 4 and the paper closes with conclusions in Section 5.

## 2 Related Work

### 2.1 Context-Aware Positioning

Context-aware Positioning has attracted significant interest by the research community over the last period. The need for this additional context was on one hand aiming to improve the energy efficiency of satellite-based navigation systems and on the other hand to provide additional knowledge into the position estimation process for indoor navigation systems in an attempt to improve the accuracy. An example of the first case is presented in [8] where the duty cycle of energy hungry GPS receivers is reduced by introducing context received from inertial sensors and positioning is performed in a hybrid manner. Information collected from inertial sensors was also proposed and used either in standalone inertial navigation systems or as complimentary context to indoor navigation solutions [9]. Moreover, for indoor navigation, many attempts have been made towards improving the accuracy of fingerprint-based positioning techniques by introducing knowledge extracted from the environment of the device to be positioned [6]. A basic limitation of fingerprint-based techniques lies on the fact that the device heterogeneity may degrade the positioning performance when the device to be positioned is different from the device that was used to collect the radio-map. Differences may arise due to varying antenna characteristics of the mobile terminals which are usually difficult to know or predict. The authors of [10] [11] have proposed the use of linear data transformation to match the characteristics of various devices by collecting only a small set of measurements using the device that is to be positioned, to calibrate a fingerprinting database which has been collected using another device. This effectively means that the type of device is a parameter that needs to be recorded during the data acquisition phase. Also, the orientation of the device is of high importance in fingerprinting positioning and in many cases measurements collected under various orientations at the same location have significant differences and therefore constitute different fingerprints [12].

Other types of context collected from audio, ambient light or other sensors or from any other built-in technology (e.g. Bluetooth) can be used to provide that extra knowledge towards a more accurate position estimation. Basically, any type of additional knowledge which would potentially give an indication about the user whereabouts can be used in conjunction with the positioning estimation process to lead to better results. For example, the authors of [13], [14], [15] propose the use of audible sound to perform or to assist the positioning estimation. The device microphone can be used to identify the rooms that the users are currently located in by matching known sounds (e.g. the sound of the washing machine). With regards to ambient light use in positioning, work in [16] reports that ambient intensity measurements have high location dependency, and they can be used for positioning with the traditional fingerprinting approach. Also, total ambient light irradiance intensity can be used to detect the proximity of a lighting source, and a location can be further resolved with the support of knowledge about the location of the lighting infrastructure. Various attempts were also reported in literature to combine context from heterogeneous radio technologies like Bluetooth, RFID etc. Such a hybrid positioning system [17] achieves better positioning

accuracy by exploiting the varying capabilities of the different technologies; that is, Wi-Fi facilitates fingerprinting positioning whereas Bluetooth—as a short-range radio technology—allows the partitioning of the indoor space as well as the large Wi-Fi radio-map by using known Bluetooth hotspots. In a similar fashion, the authors of [18] have demonstrated accuracy improvements in the fingerprint-based indoor positioning process, by imposing map-constraints into the positioning algorithms in the form of a-priori probabilities which reflect the probability of a user, to be located on one position instead of all others. These probabilities could be manually set during the off-line phase or they could be dynamically inferred during the on-line phase. An example could be that a professor is more likely to be in his office during his office hours rather than any other place on a University campus.

## 2.2 Open-Platforms for Positioning

A platform for evaluating positioning on Android devices called *Airplace* was proposed by the authors of [19]. The platform is a mobile-based network-assisted architecture which includes an RSS logger, a radio-map distribution server and a “Find Me” application which facilitates the testing of various positioning algorithms and the optimization of their settings. The authors of [20] have presented *Anyplace* [21] a free and open navigation service that relies on the abundance of sensory data on smartphones to deliver reliable indoor geolocation information. It implements a set of crowdsourcing-supportive mechanisms to handle the enormous amount of crowd-sensed data. A similar open platform is presented in [22]. This is called SmartCampusAAU and it facilitates the creation of indoor positioning systems. It includes an application and a backend that can be used to enable device- or infrastructure-based indoor positioning and a publicly available Open Data backend to allow researchers to share radio map and location tracking data. The platform relies on crowdsourcing techniques to construct radio-maps. Crowdsourcing [23], [24] leverages the positioning fingerprints collected by users using their smart-devices in order to construct and/or update the radio-map. This obviously presents various inaccuracies that need to be considered in the position estimation phase, mainly the fact that the radio-map will not be homogeneous as it contains fingerprints from a diverse set of devices. The authors of [25] have tackled this problem by collecting signal differences instead of absolute signal strength values. In the crowdsourcing process, recording the type and model of the devices is of particular importance.

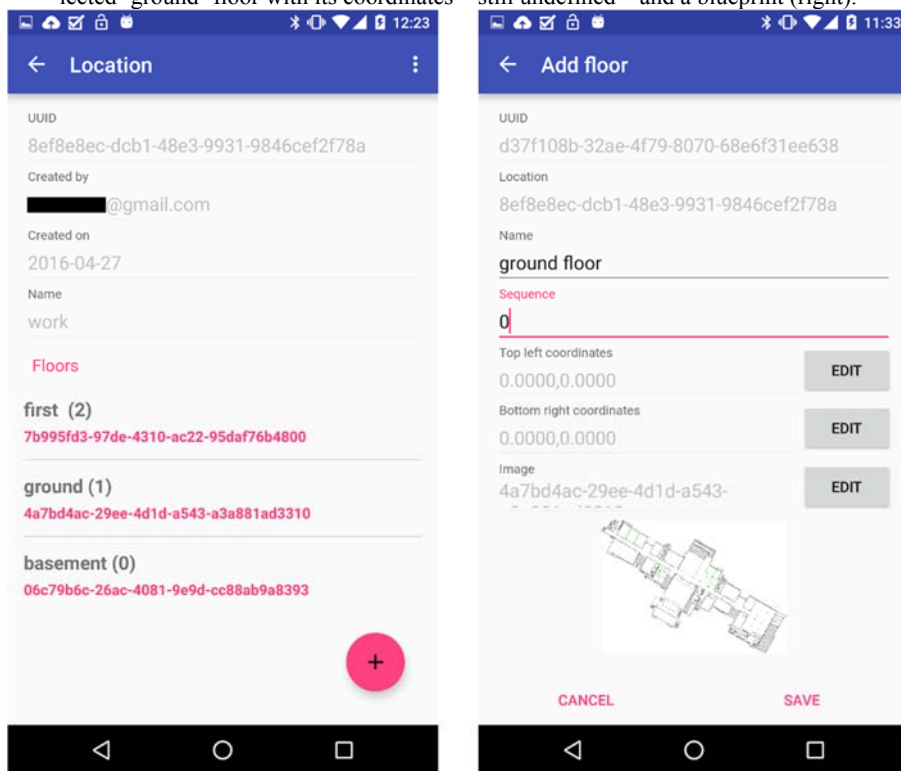
In our approach we take this open-platform concept one step further by introducing additional non-radio location-dependent context which can be openly used for developing advanced positioning algorithms, fusing together various kinds of information towards a more accurate position estimate.

### 3 Platform Design and Implementation

#### 3.1 Administration

The platform is designed as a mobile-based system which can collect, store and process data autonomously while offline. The users define their own named locations (e.g. typically a location corresponds to a building, or a group of neighboring buildings such as a campus). Each location must feature at least one floor, but possibly more (see **Fig. 1**).

**Fig. 1.** A screenshot of a view of a location and its details with 3 floors (left). Details of the selected 'ground' floor with its coordinates—still undefined—and a blueprint (right).

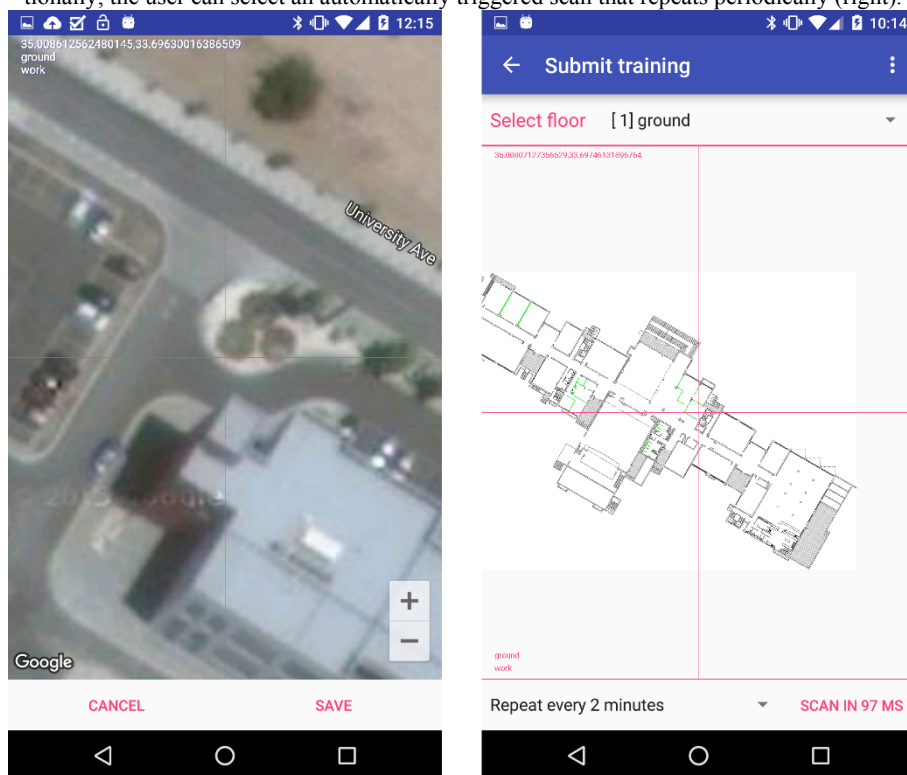


The users are asked to provide their blueprints for each floor/level, and specify the exact coordinates for the upper left and the lower right corners of the image (see right screenshot in **Fig. 1** and left screenshot in **Fig. 2**). With this information and assuming that the blueprint image is north-aligned, the system can then associate each point on the image with the corresponding geographic coordinates (i.e. latitude and longitude). This is important as it allows the users to easily specify their actual position during the training of the system, using a visual targeting system (see right screenshot in **Fig. 2**).

In the training (or offline) phase, the user collects fingerprints with the aim of generating the training data for the algorithm. Each fingerprint is associated with a well-defined location on the indoor map and more specifically on a selected floor of the

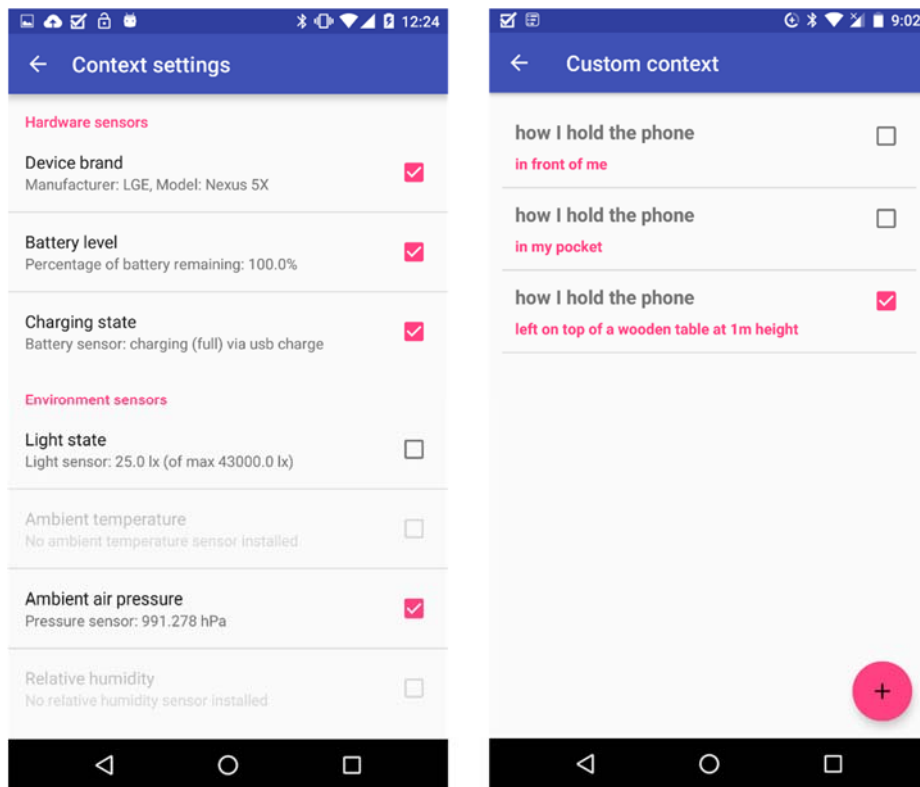
given location. The user utilizes a crosshair-like target and a draggable view of the blueprint (usually a floor map) to specify the exact position of the user at the time the fingerprint is collected. While the crosshair (pink) is fixed at the center of the view, the user is able to drag the underlying view to select her or his current position in the building (see right screenshot in **Fig. 2**). It must be assumed that the users doing the training have good knowledge of the building and are reasonably abled to navigate inside the building using the floorplan.

**Fig. 2.** Aligning the blueprint with the real world using satellite imagery (left). In the training phase, the users select their exact position—at the correct floor—and then start a new scan. Optionally, the user can select an automatically triggered scan that repeats periodically (right).



While a fingerprint typically includes only the signal strength from nearby Wi-Fi access points, the user can also specify additional context to be stored. The available context data include information that has the potential to improve the accuracy of a positioning algorithm, such as the make and model of the device, environmental data such as temperature, pressure, humidity, inertial data such as accelerometer and gyroscope readings, etc. The selection of which data to collect, is configurable via a settings screen (see left screenshot in **Fig. 3**).

**Fig. 3.** The context settings screen which allows users to preview and select the exact subset of context data they want included in the fingerprint—only sensors supported by the device’s hardware are enabled (left). A custom context screen allows the users to define extra tags and use them to annotate the collected fingerprints with context otherwise hard to infer (right).



In some cases, users would want to annotate their collected fingerprints with additional information which is otherwise hard to automatically infer using the device’s sensors. For example, a user might want to indicate whether they are holding the device at waist or eye height, hypothesizing that this might affect the received signal strength. For this reason, the app features a mechanism where users can define their own custom tags, and then select which ones to include in their training session, as needed (see right screenshot of **Fig. 3**). The users can manually select which annotations to be included during training by adding their tags, and selecting the corresponding checkboxes.

In the application (or online) phase, the user can use the system to determine her or his position in real time. Typical positioning algorithms achieve this by collecting a fingerprint from their present location, and comparing it to those stored during the training phase. The closest match (or the average of the  $K$  closest matches in the case of the  $K$ -Nearest Neighbor algorithm) is then shown as the present location on the corresponding fingerprint. A few standard algorithms are evaluated in section 4, but the system is designed to easily accommodate additional, custom algorithms.

While in its final form, the platform is envisioned to allow the use of several interchangeable algorithms selectable in runtime, in its current form the data must first be exported into a JSON-formatted file, and then be processed offline. This is described in detail in the following section. The code implementing the mobile app, as well as a console-based system for the evaluation of various fingerprint algorithms are part of a live project which is available on Github under an open-source license [26].

## 4 Evaluation

Using the proposed platform, we collected a set of samples which include the received signal strength and some additional context (i.e. battery charge level, charging status, and device model). The experiment involved 4 individuals, each using a different device from common models found in the market: Samsung Galaxy S2, Google Nexus 4, Prestigio Multipad 8, and LG Optimus L5. The measurements covered 2 floors of a medium-sized building, producing a total of 307 fingerprints. A typical fingerprint measurement includes a radio-map of 10-20 RSS measurements, along with the corresponding context data. It also includes the exact coordinates and floor of each measurement, as specified by the user. The complete, anonymized dataset used for this evaluation is openly available alongside the open-source code of this platform at Github [26]. An excerpt of the anonymized, JSON-formatted file is listed below for reference:

```
{
  "trainings": [
    {
      "uuid": "agtzf...PKtgwkM",
      "floorUUID": "a",
      "createdBy": "anonymized",
      "timestamp": 1403776198856,
      "lat": 35.008370626294614,
      "lng": 33.69661813690848,
      "context": { "batteryCharging": "false",
                  "batteryLevel": "27.0", "model": "PMP5880D",
                  "product": "asbis" },
      "measurements": [
        {"macAddress": "0", "ssid": -65 },
        {"macAddress": "1", "ssid": -66 },
        ...
        {"macAddress": "19", "ssid": -82 }
      ]
    },
    ...
  ]
}
```



To assess the quality of the dataset and the effectiveness of various fingerprinting algorithms, we implemented the following experiment: First, the collected dataset of 307 fingerprints was randomly split to two subsets: 90% for training and 10% evaluation/application. Then, each of the application fingerprints was compared to the training fingerprints, and the location was determined according to the logic of the used algorithm. The resulting error of each algorithm (i.e. distance of predicted versus actual position, in meters) is summarized in **Table 1**.

**Table 1.** Comparing the performance of a few fingerprinting algorithms. The measurements show the distance in meters from the actual target (i.e. the error) of each algorithm for a randomly chosen subset (of size 10%) of the fingerprints. All distances are in meters.

|                                | Standard deviation |          | Absolute Values |         |
|--------------------------------|--------------------|----------|-----------------|---------|
|                                | Mean               | Variance | Min             | Max     |
| Standard fingerprint algorithm | 8.7004             | 28.3425  | 1.1740          | 22.9706 |
| Same device only               | 10.1144            | 78.1406  | 2.4748          | 40.0940 |
| Similar battery only (+/- 10%) | 10.0622            | 66.9148  | 2.3564          | 37.4445 |
| K-Nearest Neighbor with K=3    | 7.6623             | 22.6163  | 0.9552          | 21.3290 |
| K-Nearest Neighbor with K=10   | 7.8376             | 21.8472  | 1.1825          | 21.0465 |

In its simplest form, the *Standard fingerprint algorithm* uses the Euclidean distance to measure the distance among individual readings in the radio-map, as described in [27]. When comparing fingerprints, missing access point readings were assumed to have a zero RSS. The *Same device only* variant uses the exact same mechanism, but filters training fingerprints to select only those that were generated by the same device model. Similarly, the *Similar battery only* variant filters fingerprints to select those only that were generated by devices which (at the time of the training) had similarly charged batteries. The last two variants, refer to using the *K*-Nearest Neighbor algorithm with *K* equals 3 and 10 respectively. In this case, the algorithm first identifies the best *K* matches (i.e. best 3 or 10 accordingly) using the standard fingerprint algorithm, and then decides the inferred position to be the center of those best matches.

Using the samples collected in our experiment, we applied each of the algorithms described above, and measured their perceived accuracy. **Table 1** lists the performance of each of the algorithms in terms of the mean distance and the variance for all tests, using standard deviation calculations on the resulting measurements of the distance error. For reference, the table also lists the minimum and the maximum distances measured. All measurements in **Table 1** are in meters.

Studying these measurements reveals that the standard fingerprint algorithms works quite effectively and precisely already in its simplest form. Utilizing additional context information such as the device model or the battery charge level did not appear to improve the accuracy of the algorithm, which can be partly explained by the fact that the training phase was rather dense (with multiple fingerprints in each room, for each user and each device model). This is also evident in the fact that the *K* Nearest Neighbor-based techniques have performed better than all others (especially the *best of 3* variant), indicating that the fingerprints are densely covering the building's area.

The experiment has shown that the presented platform provides a reliable and convenient mechanism for collecting fingerprints that are accurate and include several context data. We argue that this will assist in creating an open ecosystem where multiple shared datasets can be utilized to evaluate new fingerprint-based algorithms, and/or fine-tune existing ones.

## 5 Conclusions

Indoor positioning is an increasingly important topic, with both a social and financial impact. Most commonly, fingerprint-based algorithms are used to enable indoor positioning, where the fingerprints consist primarily of radio-maps. Nevertheless, with the advancement of sensory technology on new smart-phone devices, algorithms are increasingly dependent on additional context information to increase their accuracy. Even though many such algorithms have been developed, very few have been tested or were applied to real-world conditions.

In this paper we present an open platform which enables researchers and practitioners to easily perform their own experiments using fingerprint datasets which fuse together radio-specific context with additional context that could potential offer better positioning estimates. Users are enabled to quickly set up their locations, floors etc. and then perform the training. The collected data can be easily exported as JSON-formatted data which can then be used for offline assessment of the accuracy of arbitrary algorithms. Additionally, generated datasets can be shared with the scientific community. In a preliminary evaluation, we have shown how the collection of such data can enable testing arbitrary fingerprinting algorithms, as well as variants of them, in a straightforward and convenient way.

In this ongoing project, we envision to further enhance the mobile platform and integrate it with a cloud-based system for storing, sharing and making datasets openly available. We also aim at enabling the collection of additional type of context, as well as covering more extensive time periods and more device models, while in the process producing exemplar open research datasets.

## References

- [1] D. Macagnano, G. Destino and G. Abreu, "Indoor positioning: A key enabling technology for IoT applications," in *IEEE World Forum on Internet of Things (WF-IoT)*, Seoul, 2014.
- [2] I. Jami, N. M. F. N. M. Ali and R. F. Ormondroyd, "Comparison of methods of locating and tracking cellular mobiles," in *IEE Colloquium on Novel Methods of Location and Tracking of Cellular Mobiles and Their System Applications (Ref. No. 1999/046)*, London, 1999.

- [3] T. S. Rappaport, J. H. Reed and B. D. Woerner, "Position location using wireless communications on highways of the future," *IEEE Communications Magazine*, vol. 34, no. 10, pp. 33-41, 1996.
- [4] P. B. a. V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *19th Annual Joint Conference of the IEEE Computer and Communications Societies*, Tel Aviv, 2000.
- [5] V. Honkavirta, T. Perala, S. Ali-Loytty and R. Piche, "A comparative survey of WLAN location fingerprinting methods," in *6th Workshop on Positioning, Navigation and Communication (WPNC)*, Hannover, 2009.
- [6] M. Raspopoulos, B. Denis, M. Laaraiedh, J. Dominguez, L. De Celis, D. Slock, G. Agapiou and J. S. S. Stephan, "Location-dependent information extraction for positioning," in *International Conference on Localization and GNSS*, Starnberg, 2012.
- [7] L. Sangwoo, C. Bongkwan, K. Bonhyun, R. Sanghwan, C. Jaehoon and K. Sunwoo, "Kalman filter-based indoor position tracking with self-calibration for RSS variation mitigation," *International Journal of Distributed Sensor Networks*, vol. 11, no. 8, p. 180, 2015.
- [8] R. Jurdak, P. Corke, D. Dharman and G. Salagnac, "Adaptive GPS Duty Cycling and Radio Ranging for Energy-efficient Localization," in *The 8th ACM Conference on Embedded Networked Sensor Systems (SenSys 2010)*, Zurich, Switzerland, 2010.
- [9] R. Harle, "A Survey of Indoor Inertial Positioning Systems for Pedestrians," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1281-1293, 2013.
- [10] M. Raspopoulos, C. Laoudias, L. Kanaris, A. Kokkinis, C. G. Panayiotou and S. Stavrou, "3D Ray Tracing for device-independent fingerprint-based positioning in WLANs," in *9th Workshop on Positioning Navigation and Communication (WPNC)*, Dresden, 2012.
- [11] M. Raspopoulos, C. Laoudias, L. Kanaris, A. Kokkinis, C. G. Panayiotou and S. Stavrou, "Cross device fingerprint-based positioning using 3D Ray Tracing," in *8th International Wireless Communications and Mobile Computing Conference (IWCMC)*, Limassol, 2012.
- [12] D. Su, Z. Situ and I. W.-H. Ho, "Mitigating the antenna orientation effect on indoor Wi-Fi positioning of mobile phones," in *IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Hong Kong, 2015.
- [13] A. Mandal, C. V. Lopes, T. Givargis, A. Haghghat, R. Jurdak and P. Baldi, "Beep: 3D indoor positioning using audible sound," in *Second IEEE Consumer Communications and Networking Conference CCNC*, Las Vegas, 2005.

- [14] A. Madhavapeddy, D. Scott and R. Sharp, "Context-Aware Computing with Sound," in *5th International Conference on Ubiquitous Computing*, Seattle, 2003.
- [15] C. Joe, L. Yip, J. Elson, H. Wang, D. Maniezzo, R. E. Hudson, K. Yao and D. Estrin, "Coherent Acoustic Array Processing and Localization on Wireless Sensor Networks," *Center for Embedded Network Sensing*, vol. 91, no. 8, pp. 1154 - 1162, 2003.
- [16] J. Liu, Y. Chen, A. Jaakkola, T. Hakala, J. Hyyppa, L. Chen, R. Chen, J. Tang and H. Hyyppa, "The uses of ambient light for ubiquitous positioning," in *IEEE/ION Position, Location and Navigation Symposium (PLANS2014)*, Monterey, 2014.
- [17] A. Baniukevic, C. S. Jensen and H. Lu, "Hybrid Indoor Positioning with Wi-Fi and Bluetooth: Architecture and Performance," in *IEEE 14th International Conference on Mobile Data Management (MDM)*, Washington, DC, 2013.
- [18] A. Kokkinis, M. Raspopoulos, L. Kanaris, A. Liotta and S. Stavrou, "Map-aided fingerprint-based indoor positioning," in *IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, London, 2013.
- [19] C. Laoudias, G. Constantinou, M. Constantinides, N. S. D. Zeinalipour-Yazti and C. Panayiotou, "The Airplace Indoor Positioning Platform for Android Smartphones," in *2012 IEEE 13th International Conference on Mobile Data Management*, Bengaluru, Karnataka, 2012.
- [20] "Demonstration Abstract: Crowdsourced Indoor Localization and Navigation with Anyplace International conference on Information processing in sensor networks," in *IPSN'14, IEEE Press*, Berlin, Germany, 2014.
- [21] [Online]. Available: <https://anyplace.cs.ucy.ac.cy/>.
- [22] R. Hansen, B. Thomsen, L. L. Thomsen and F. Stubkjær, "SmartCampusAAU -- An Open Platform Enabling Indoor Positioning and Navigation," in *14th International Conference on Mobile Data Management*, Milan, 2013.
- [23] P. Mazumdar, V. J. Ribeiro and S. Tewari, "Generating indoor maps by crowdsourcing positioning data from smartphones," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Busan, 2014.
- [24] C. Wu, Z. Yang and Y. Liu, "Smartphones Based Crowdsourcing for Indoor Localization," *IEEE Transactions on Mobile Computing*, vol. 14, no. 2, pp. 444 - 457, 2015.
- [25] C. Laoudias, D. Zeinalipour-Yazti and C. G. Panayiotou, "Crowdsourced Indoor Localization for Diverse Devices through Radiomap Fusion," in *International Conference on Indoor Positioning and Indoor Navigation*, Montbeliard-Belfort, 2013.

- [26] N. Paspallis, "Context-Aware Indoor Positioning System," [Online]. Available: <https://github.com/nearchos/CAIPS>. [Accessed 03 Oct. 2016].
- [27] A. Varshavsky and S. Patel, "Location in Ubiquitous Computing," in *Ubiquitous Computing Fundamentals*, J. Krumm, Ed., Boca Raton, Chapman and Hall/CRC, 2009, pp. 285-320.