Experimental Decomposition of the Performance of Fingerprinting-based Localization Algorithms

Filip Lemic, Arash Behboodi, Vlado Handziski, Adam Wolisz
Telecommunication Networks Group (TKN), Technische Universität Berlin (TUB)
{lemic, behboodi, handziski, wolisz}@tkn.tu-berlin.de

Abstract—Despite their popularity, the current praxis of comparative experimental evaluation of fingerprinting-based localization algorithms is lacking rigor, with studies typically following an ad-hoc evaluation process and focusing on black-box comparison of complete algorithms. In this paper we present a systematic benchmarking methodology that is focused on gaining fine-grained insight about the relative contributions of the individual phases of the fingerprinting-based localization algorithms to their overall performance. To this end, we decompose the localization algorithms in common phases (collection of raw measurements, creation of fingerprints, pattern matching and post-processing) and systematically assess the performance of different procedures that can be applied in each of these phases. We illustrate the application of the proposed methodology using a comprehensive experimental case-study of 3 WiFi fingerprinting algorithms with 4 raw RSSI collection procedures, 3 fingerprint creation and pattern matching procedures, 4 different post-processing procedures in 3 testbeds and 4 evaluation scenarios, resulting in 36 individual experiments. The results demonstrate that in the evaluated scenarios, a lower number of WiFi APs and rather simple fingerprint creation and pattern matching can achieve better performance in terms of location accuracy than more sophisticated alternatives. The results also show that post-processing steps like k-Nearest Neighbours (kNN) procedure are indeed effective in reducing the localization error variability and extremes, thus increasing the stability of the location estimation.

Index Terms—indoor localization, WiFi fingerprinting algorithms, WiFi beacon packets, Received Signal Strength Indicator, fingerprinting phases

I. INTRODUCTION

The growing interest for indoor location-based services, like tracking equipment and material (e.g. in hospitals and factories) or people (e.g. early responders in emergencies, medical staff in hospitals, customers in shopping malls) has created strong demand for effective and affordable indoor localization solutions. Fingerprinting-based localization, leveraging the widely deployed Wireless Fidelity (WiFi) infrastructure, is one of the most popular classes of indoor localization approaches. Using WiFi signals as fingerprinting sources is also attractive due to the capability of Radio Frequency (RF) waves in this frequency to penetrate walls and obstacles and the relatively good resistance to external interference as a result of the use of robust coding and modulation [1]. The above mentioned advantages of WiFi-based fingerprinting led to a plethora of different algorithm proposals, [2]–[4] being just few examples. Despite their popularity and importance, the current praxis of comparative experimental evaluation of WiFi-based fingerprinting solutions is lacking rigor, with studies typically following an ad-hoc evaluation process and focusing on black-box comparison of complete algorithms. Usually, the algorithms are presented and evaluated as monolithic solutions, although several well-defined phases can be identified: collection of raw data, creation of fingerprints, matching of fingerprints and training data and post-processing for which specific procedures can be separately defined. Thus, the existing evaluation approaches fail to offer deeper insight into the relative contribution of these individual phases to the overall performance of the solutions.

In this paper we apply a methodology for systematic experimental evaluation of the performance of WiFi-based fingerprinting solutions that is based on a more general benchmarking approach for objective evaluation of indoor localization algorithms, developed as part of the EVA RILOS project [5]. We show how this methodology can be effectively used to study the relative contribution of various procedures used in the individual algorithm phases on the overall performance, confirming and extending the insights of earlier, less systematic attempts in this direction [1], [6], [7].

We illustrate the application of the proposed methodology using a comprehensive experimental case-study of 3 WiFi fingerprinting algorithms with 4 raw RSSI collection procedures, 3 fingerprint creation and pattern matching procedures, 4 different post-processing procedures in 3 testbeds and 4 evaluation scenarios, resulting in 36 individual experiments. The results confirm the relatively large impact of the procedures in the early phase of collection of raw RSSI measurements, something that is usually only marginally considered in the design of many fingerprinting algorithms. They show that, somewhat counter-intuitively, a lower number of WiFi APs and simpler fingerprint creation and pattern matching procedures can sometimes match and even outperform more sophisticated alternatives, in terms of location accuracy. They also highlight the importance of the post-processing and filtering step, like kNN, in reducing the localization error variability and extremes.

These results illustrate the importance of customized selection of procedures for the individual phases of fingerprinting algorithms, in accordance to the specific conditions in different deployment environments and in light of the relevant performance metrics. Although the presented study is just a snapshot, we believe that when applied to a wider class of procedures and their combinations, our evaluation methodology can lead to significant improvement in the fundamental understanding
of the process of composing efficient fingerprinting algorithms for specific deployment scenarios.

The rest of this paper is structured as follows. Section II presents the decomposition of fingerprinting algorithms into main phases and provides detailed description of the procedures used in each of them. Section III describes the methodology used for objective comparison of the performance of fingerprinting algorithms, including a description of the experimental environments. In Section IV we give an overview of the evaluation scenarios. Section V presents the evaluation results and discusses the influence of the different phases to the overall performance, while Section VI concludes the paper.

II. FINGERPRINTING BASED LOCALIZATION USING WiFi BEACON PACKETS RSSI MEASUREMENTS

Fingerprinting algorithms can be typically divided in three phases, while the execution of the fingerprinting procedure has two distinguished steps (Figure 1). The first step, called training, is executed off-line in order to obtain a footprint of wireless environment. In this step, the localization area is divided in a number of segments (cells). Each cell is scanned certain number of times for raw data (phase I), and a fingerprint of each cell is created (phase II) following some methodology for processing the collected raw data. Using the obtained training fingerprints, the training database is created and stored on the localization server. The second step, known as runtime (on-line) step, is executed with participation of the localization service user and consists of three individual phases. Two of them are mirroring the phases of the first step: a number of scans are first created by the user at the unknown position, and the runtime fingerprint for the position is created following the same procedure as in the training step. On the server’s side, the runtime fingerprint is compared with the training dataset following some predefined matching procedure (phase III). The output of the matching phase is then typically post-processed using a procedure like the kNN method. Actions within each phase are specified by so called procedures, in the following we will discuss examples of them.

![Figure 1: Phases of WiFi fingerprinting algorithms](image)

A. Collection of Raw RSSI Measurements

The first phase of fingerprinting algorithms is constituted by the collection of raw data used for collecting Received Signal Strength Indicator (RSSI) measurements. While sources of raw data can be different, we will, for simplicity, talk only about RSSI measurements from WiFi beacon packets as most popular. Three basic decisions define this phase: The locations of measurements, the number of measurements that should be collected in each location, and the set of beacon sources (Access Points (APs)) which are considered. Let \( N \) be the number of the environmental scans taken at one measurement point. During each scan the vector of RSSI measurements from APs is collected. By repeating the measurements at one location \( N \) times a matrix of RSSI measurements from different APs is created. Similar procedure is followed in both the training and runtime steps, with \( N \) being usually smaller in the runtime step due to time and energy constraints. To limit the size of the raw dataset and to improve the robustness, different methods can be used to select the set of contributing APs. Typical examples are:

1) **All Visible APs:**
   In this case not only APs being under control of the indoor localization provider and thus expected to operate continuously with stable parameters, but also all available APs are included (e.g. APs installed in neighboring buildings).

2) **All Visible APs with Threshold:**
   In this case only RSSI measurements above certain threshold are used as raw data. The goal of the scoping is to reduce the variability of RSSI measurements in time by discarding those below certain threshold. In this paper we will consider two such threshold values: -90 dBm and -80 dBm.

3) **Dedicated APs:**
   The third approach in collection of raw RSSI measurements that we evaluate is selective collection of RSSI measurements only from “dedicated” set of APs. Albeit their primary function is obviously providing wireless connectivity, this set is assumed to be “under control” of the localization service provider (or her contractual partners). Therefore the stability of their operation, and high signal strength can be expected.

B. Fingerprint Creation

In the fingerprint creation phase certain characteristic values are derived from the raw RSSI measurements according to some procedure. Note that selection of this procedure is independent of the decisions made in Phase I. These procedures might relate to time series of measurements from a single AP, or involve whole vectors of APs. For simplicity we will further present examples of the fingerprints relating to each AP separately, listed in the sequence of increasing computational complexity.

1) **Averaged RSSI Vectors:**
   A simple, popular procedure [1] is computing an average value of the RSSI measurements obtained from each AP used for localization. The fingerprint is an average value of the RSSI measurements obtained from each AP used for localization in both training and runtime steps, where \( K_{r,t} \) is the length of the vector. Let \( \mu_{t,m} = [RSSI_{t,1},...,RSSI_{t,k},...,RSSI_{t,K_{t,r}}] \) be the vector of averaged RSSI values from each AP \( i \) obtained in training step at point \( m \in 1,...,M_t \), i.e. training fingerprint. In the same manner, let \( \mu_r = [RSSI_{r,1},...,RSSI_{r,k},...,RSSI_{r,K_{r,r}}] \) be the vector of averaged RSSI values from each AP \( i \) obtained in runtime...
This well known pattern matching procedure \cite{4} uses the complexity, we present them sequentially. Our choice of three sample ones and from the point of view of computational various different procedures for pattern matching we select procedure is then reported as an estimated location. From the runtime fingerprint according to a given pattern matching similarities between a training dataset and a runtime fingerprint. 

3) Multivariate Gaussian (MvG) Distributions of RSSIs: Another recently proposed sample fingerprint creation procedure uses the Multivariate Gaussian (MvG) distributions of RSSI measurements from each AP used for localization \cite{4}. The procedure assumes that the RSSI measurements from different APs are distributed according to the MvG distribution. In other words, the distribution of the RSSI measurements from each AP at one cell can be written as $\mathcal{N}(\mu, \Sigma)$, where $\mu$ is the vector of averaged RSSI values from each AP used for localization and $\Sigma$ is the covariance matrix describing the relation between RSSI measurements from different APs.

C. Pattern Matching

The pattern matching phase is generally used for capturing similarities between a training dataset and a runtime fingerprint \cite{9}. The training fingerprint that is most similar to a runtime fingerprint according to a given pattern matching procedure is then reported as an estimated location. From the various different procedures for pattern matching we select three sample ones and from the point of view of computational complexity, we present them sequentially. Our choice of presented procedures is not completely free, while it has to be constrained dependent on the fingerprinting creation.

1) Euclidean distance:
This well known pattern matching procedure \cite{4} uses the Euclidean distance (ED) between training fingerprint at the cell $m$ and the runtime fingerprint and it is given as:

$$D_E(\mathbf{X}_{t,m}, \mathbf{X}_r) = \|\mathbf{X}_{t,m} - \mathbf{X}_r\|$$  \hspace{1cm} (1)

where $\mathbf{X}_{t,m}$ and $\mathbf{X}_r$ are fingerprint vectors in training and runtime step, respectively. The cell with the smallest distance (also called smallest weight) is reported as an estimated location. In this paper we use ED as pattern matching procedure for the Averaged RSSI fingerprints.

2) Pompeiu-Hausdorff Distance:
This recently proposed pattern matching procedure uses Pompeiu-Hausdorff (PH) metric for capturing similarities between fingerprints in training dataset and runtime fingerprint \cite{8}. The Pompeiu-Hausdorff (PH) distance between two sets is given as follows:

$$D_{PH}(\mathbf{X}_{t,m}, \mathbf{X}_r) = \max_{x \in \mathbf{X}_{t,m}} \min_{x_r \in \mathbf{X}_r} d(x_{t,k}, x_{r,k})$$  \hspace{1cm} (2)

where $d(x_{t,k}, x_{r,k})$ is the Euclidean Distance measurement between elements of the runtime fingerprint $\mathbf{X}_r$ and training fingerprint $\mathbf{X}_{t,m}$ at point $m$. The training point with the smallest PH distance with the runtime fingerprint is reported as an estimated location. In this paper we use PH as pattern matching procedure for RSSI quantile fingerprints.

3) Kullback-Leibler Distance:
Another recent pattern matching procedure uses the Kullback-Leibler (KL) distance metric between training fingerprint at the point $m$ and the runtime fingerprint for capturing the similarities between them \cite{4}. In this procedure training and runtime fingerprints are Probability Density Functions (PDFs) given by $p_{t,m}$ and $p_r$, respectively. We apply KL distance as pattern matching for MvG fingerprints. The Kullback-Leibler (KL) distance metric between the probability distributions in this case is given as:

$$D_{KL}(p_{t,m}, p_r) = \frac{1}{2}((\mu_r - \mu_{t,m})^T(\Sigma_r)^{-1}(\mu_r - \mu_{t,m}) + tr((\Sigma_r)^{-1}\Sigma_{t,m} - I) - ln|\Sigma_r|^{-1}\Sigma_{t,m}|)$$  \hspace{1cm} (3)

where $tr(\cdot)$ denotes the trace of a matrix (sum of its diagonal elements) and $I$ is the identity matrix. This pattern matching procedure reports the cell with the smallest KL distance to the runtime fingerprint as an estimated location.

D. Post-processing

A post-processing method, like the application of kNN, is usually the last phase of the fingerprinting procedure. The input from previous phase is the location estimate, which can be easily extended to a set of best estimates. Namely, instead of using the best match between training dataset and runtime fingerprint, one can use the set of $k$ best matches. Based on this set of training fingerprints considered as best matches, different kNN methods for estimating the final location can be applied. From the numerous possible kNN methods for the simplicity reasons we select two well known and widely used ones and use them as representatives for evaluating the influence of kNN methods.

1) Non-weighted k-Nearest Neighbors:
Non-weighted kNN methods give the same weight to each of the $k$ fingerprints reported as best matches with the runtime fingerprint. We use $3$ or $4$ as parameters of non-weighted kNN method for evaluation of influence of kNN method on the accuracy of indoor fingerprinting.

2) Weighted k-Nearest Neighbors:
Weighted kNN neighbors assign different weights to the training fingerprints reported as best matches in the previous phase. We use two different sets of weights, namely for $k = 3$ weights are $[4/7, 2/7, 1/7]$ and for $k = 4$ weights are $[8/15, 4/15, 2/15, 1/15]$.

III. BENCHMARKING OF INDOOR LOCALIZATION

In this section we present our approach for addressing the problem of objective comparison of fingerprint-based indoor localization algorithms by following the comprehensive benchmarking methodology developed in the EUVARILOS project. The EUVARILOS Benchmarking Handbook (EBH) offers a set of guidelines for evaluation and comparison of different indoor localization solutions \cite{5}. The EBH promotes the usage of multiple metrics for the performance evaluation like
accuracy, latency, energy efficiency, interference robustness, etc. We have selected the geometrical and room level accuracy metrics as the focus of the study presented in this work. Geometrical accuracy implies the Euclidean error distance between a reference and an estimated location, while room accuracy evaluates the correctness of the estimation on room/space granularity. The EBH also promotes the use of well defined evaluation scenarios and environments in which the comparative evaluation is performed. From the set of generic scenarios specified in the EBH, we describe and instantiate three office scenarios and an open space scenario, all with minimized effects of external interference. We perform our experiments in three testbeds, described in the text below.

A. TWIST Testbed Description

The TKN Wireless Indoor Sensor network Testbed (TWIST) testbed is located at the 2\textsuperscript{nd}, 3\textsuperscript{rd} and 4\textsuperscript{th} floor of the Telecommunication Networks Group (TKN) building in Berlin [10], [11]. According to the EBH, the TWIST testbed environment can be characterized as “Big” with “Brick walls”, i.e. more than 1200 m\textsuperscript{2} area (approx. 30 x 15 m, 3 floors) with more than 50 rooms. The footprints of each floor of the testbed are given in Figure 2. Black dots in the figure present locations of the dedicated APs, while red dots present the location of training points used in the fingerprinting procedure. The reason for non-uniformity of locations of training points in this testbed lies in the fact that we reused the known locations of low-power sensor nodes in the testbed [10].

The dedicated wireless APs used for localization are TL-WDR4300 wireless router, with a fixed channel allocation scheme set on channel 11 (2462 MHz), with the transmission power of dedicated APs set to 20 dBm (100 mW). All experiments are performed during weekend afternoons, in an environment with minimized interference. The wireless environment was monitored using WiSpy 2.4x spectrum scanners, and all samples taken in the presence of interference above a threshold of -80 dBm were repeated. As a client’s device MacBook Pro notebook with the AirPort Extreme Network Interface Card (NIC) was used.

B. w-iLab.t I Testbed Description

The w-iLab.t I wireless testbed used for indoor localization benchmarking purposes is located at De Zuiderpoort in Ghent, Belgium. According to the EBH, the testbed area used for benchmarking is characterized as “Medium” with “Plywooden walls”, i.e. more than 800 m\textsuperscript{2} office area (approx. 40 x 20 m). The footprint of the w-iLab.t I testbed is given in Figure 3, together with locations of training points (red dots) and locations of dedicated APs (black dots) used for indoor localization fingerprinting procedure.

The intermediate nodes (called iNodes) are Alix 3C3 devices running Linux. These are mini PCs equipped with two IEEE 802.11 a/b/g interfaces. They have been configured to a fixed channel allocation scheme set on channel 11 (2462 MHz), with the transmission power of dedicated APs set to 20 dBm (100 mW).

C. w-iLab.t II Testbed Description

The third testbed we use for benchmarking purposes is the w-iLab.t II wireless testbed [12]. The testbed is located in Ghent and it is a part of Future Internet Department of iMinds. With the size of more than 1000 m\textsuperscript{2} (50 x 25 m) and according to the EBH, this testbed can be characterized as an “Open-space” environment of the size “Big”. The footprint of w-iLab.t II testbed is given in Figure 4. Black dots represent locations of the APs used for localization, while red dots present locations of training points for the fingerprinting procedure. Other objects in figure represent the locations of obstacles in the environment. Obstacles are mostly made of metal so a lot of shielding and reflection in the environment is expected.

The APs used for localization are Zotac embedded PCs with IEEE 802.11n wireless cards. The same type of device is used as a client device for localization. Other parameters of the experiment are similar as in the TWIST testbed. The transmission power of the dedicated APs is set to 20 dBm (100 mW) and the fixed channel allocation scheme is set on
channel 11 (2462 MHz). The environment is shielded so there is no external interference.

IV. BENCHMARKING SCENARIOS

In the following we present four evaluation scenarios and their instantiations in the testbeds. For each of those scenarios we made a certain number of fingerprints both in training and runtime steps. One fingerprint of the environment in the training step consists of 40 scans of the RF environment, where each one is made by scanning all the available WiFi channels (1 to 11) for 1 s. The same procedure is used for gathering the runtime fingerprints, but here one fingerprint consists of only 10 scans of the WiFi environments. This reflects the more stringent constraints on the latency and power consumption in the runtime step.

A. First Scenario Instantiation

According to the EBH, the first scenario can be characterized as “Small size office environment”. Due to that, this scenario is instantiated on the 2nd floor of TWIST testbed. The testbed environment has been scanned for fingerprints as presented in Figure 2 and the training dataset has been created. The training dataset for this scenario then consists of 41 training points and each point consists of 40 scans of the RSSI measurements. Some APs are not visible at some training points or some scans. In this case, these RSSI measurements are given a default RSSI value (-100 dBm). The measurement points of the second scan of the environment, used as the runtime dataset, are shown in Figure 5.

B. Second Scenario Instantiation

Second scenario can be characterized as the “Medium size office scenario”. Scenario is instantiated in the w-iLab.t I testbed in Ghent. The training dataset consists of 56 training fingerprints, as presented in Figure 3. The set of 20 measurement points, used as the evaluation points in the runtime step of fingerprinting, are presented in Figure 6. Some APs are possibly not visible on all measurement points and in this case the RSSI measurements from these points are given the default values of -100 dBm.

C. Third Scenario Instantiation

The third scenario is characterized as a “Big size office environment” and it is instantiated on 2nd, 3rd, and 4th floor of TWIST testbed. Each floor is supplied with 4 dedicated APs, so altogether 12 APs are set up in the localization area and used as the dedicated localization APs. The training dataset is filled with the scans of the environment at each training point, and finally it consist of 123 fingerprints. Same as for the previous scenarios, each training point is scanned 40 times. The RSSI measurements that are not visible at particular point are given a default value of -100 dBm. The runtime dataset consists of the measurements at the same positions as in the scenario 1, but extended on the 3rd and 4th floors. The runtime points of the 2nd floor are depicted in Figure 5, while the points on other two floor have the same locations in terms of x and y coordinate, while coordinate z differs at each floor.

D. Fourth Scenario Instantiation

According to the EBH, the fourth scenario can be characterized as the “Big open-space” environment, and it is instantiated in the w-iLab.t II testbed in Ghent. Only 4 dedicated APs were used for localization. The training dataset consists of 100 fingerprints with locations given in Figure 4. The runtime dataset consists of 27 fingerprints equally distributed in the testbed environment, shown with dots in Figure 7. The rest remains as in the previous scenarios.

V. RESULTS OF ALGORITHMS COMPARISON

This section presents a comparison of the algorithms performance in terms of the geometrical and room level accuracy of the location estimation. Firstly, we present the influence of different procedures for raw RSSI measurement collection on the evaluation metrics. Namely, we estimate the final location using different procedures for collection of raw RSSI measurements, while fixing the fingerprints creation and pattern
matching procedures. Due to the large number of possible combinations of fingerprint creation and pattern matching procedure, in this study we have reduced the number of evaluated alternatives to several natural combinations. For combinations of fingerprint creation and pattern matching procedures we choose ones in which both fingerprint creation and pattern matching have the same complexity level. In other words, we match the least complex fingerprint creation with the least complex pattern matching procedure. Finally, we evaluate the influence of different kNN procedure on the accuracy of indoor fingerprinting.

We perform the evaluation of collection of raw RSSI measurements in the office scenarios in TWIST testbed, because this environment is not shielded and number of uncontrollable APs, beside the dedicated APs used for localization, are expected. Secondly, for all presented scenarios, we compare the effect of different procedures of fingerprint creation and pattern matching on the evaluation metrics, for fixed procedure of the raw RSSI collection. Finally, we apply different kNN procedures on the results of pattern matching to evaluate the final performance of different indoor WiFi based fingerprinting algorithms.

A. Results using Different Procedures for Raw RSSI Collection

The influence of procedures for collection of raw RSSI measurements on evaluation metrics in scenarios 1 and 3 is given in Figure 8 and Figure 9, respectively. Namely, we collected the raw RSSI measurements from all APs visible in the environment and filtered them in accordance to the respective raw RSSI collection procedure. The other two scenarios have been excluded from this evaluation, since only dedicated APs are present there, i.e. no uncontrolled APs are visible.

In this manner, we used the data from the same experiment for all raw RSSI collection procedures. On the differently obtained raw RSSI measurements we applied the same fingerprint creation and pattern matching procedures in order to estimate the location, so the influence of the procedure of raw RSSI collection is captured in the final location estimation. Summary results on the average localization error and room level accuracy in scenarios 1 and 3 are given in Table I and Table II, respectively.

The results of our experiment show that the best performance of different fingerprinting algorithms is achieved using RSSI measurements from dedicated APs, meaning that in both scenarios in the testbed environment and for all given fingerprint creation and pattern matching procedures at least similar results are achieved using dedicated APs, compared to using all visible APs, with or without a threshold.

B. Results using Different Fingerprint Creation and Pattern Matching Methods

A comparison of different fingerprint creation and pattern matching blocks for all scenarios is presented in Figure 10. For the collection of raw RSSI measurements only dedicated localization APs are used, due to the best performance of this procedure as shown previously. The average localization error of the KL Distance of MvG Distributions of RSSIs algorithm is 2.77, 17.57, 6.84 and 17.63 m for scenarios 1, 2, 3 and 4, respectively. Furthermore, the average localization error for...
the ED of Averaged RSSI Vectors is significantly smaller for the office scenarios, namely 2.21, 2.67 and 2.09 m for scenarios 1, 2 and 3 respectively. As for the open space scenario, the localization error of ED Distance of Averaged RSSI Vectors is 14.9 m. Finally, the PH Distance of RSSI Quantiles performs similar as ED Distance of Averaged RSSI Vectors, in terms of the average localization error in office scenarios, namely, the average error equals 2.02 and 2.75 and 2.05 m for scenarios 1, 2 and 3, respectively. For the open-space scenario, scenario 4, the average localization error is 8.01 m, which is an improvement of more than 6 m or 40% in comparison to the other two algorithms. Table III presents summary results of performance of the presented localization algorithms. The obtained accuracies show that at least for the selected fingerprint creation and pattern matching procedures, more complexity does not necessarily translate into increased location estimation accuracy. To a certain degree counter intuitive, less and medium complex procedures achieved better performance in presented environments and scenarios.

C. Results using Different k-NN Methods

This section presents the results of evaluation using different kNN procedures. Namely, we applied two different kNN procedures, weighted and non-weighted, with two sets of parameters, as described previously in the paper. Tables IV, V, VI and VII summarize the evaluation results for scenario 1, 2, 3 and 4, respectively. Due to the space limitations, we present the CDF of localization errors only for scenario 1. Namely, Figure 11 depicts the influence of different kNN procedures on the average localization error, using different procedures for fingerprint creation and pattern matching, and only dedicated APs in the raw RSSI collection.

From the results obtained in 4 different scenarios, we can conclude that kNN procedure, when applied on the KL Distance of MvG Distribution of RSSIs procedures for fingerprint creation and pattern matching, generally increases the localization error, which is caused by inaccurate set of best estimates, save the best one. However, for the other two fingerprint creation and pattern matching procedures applying the kNN procedure generally reduces the variability of the localization error.
Figure 9: Difference in localization error due to different procedures for collection of RSSI measurements.

The results suggest that the variances and maximum values of localization errors are reduced when kNN procedure is applied on those two fingerprint creation and pattern matching procedures. However, accuracy of WiFi based fingerprinting algorithms, when different kNN procedures are applied, seems to depend on the type and size of the environment. In the office scenarios, it seems that applying the kNN in the small environment does not improve the accuracy of fingerprinting algorithms, and the improvement is more visible when kNN procedures are applied in the environments with bigger size.

Finally, in all scenarios and for all fingerprint creation and pattern matching procedures the weighted 4-NN procedure generally gave the best results compared to the other kNN procedures. Depending on the environment and fingerprint creation and pattern matching methods, these results are comparable or better that the results obtained without applying kNN procedure.
In this work we presented objective comparison of selected WiFi fingerprinting algorithms following a systematic benchmarking methodology, developed as part of the EVARILOS project and formalized in the EVARILOS Benchmarking Handbook. We evaluated the contributions of procedures in the individual phases on the final performance of fingerprinting-based localization algorithms. We illustrated how the method for raw RSSI collection influences the final accuracy and showed that skipping the collection only to the set of dedicated APs gives the best results in the presented environments and scenarios. We compared three different methods for fingerprint creation and pattern matching and showed in our experiments that ones with rather low or medium computational complexity achieved better performance results. Moreover, we evaluated the influence of post-processing on the example of kNN procedures, showing that applying kNN procedure in fingerprinting...
decreases the variability and maximum localization errors, i.e. improving the stability of fingerprinting algorithms. We showed that this influence, however, depends on the type of the deployment environment. The accuracy of algorithms with and without kNN is comparable in the small environments, while the improvement when applying the kNN is more visible in environments with bigger size. Finally, based on the results of our experiments in the three different office scenarios 1-3, we observe that in the similar environments, in terms of the environment size, type and density of APs, the evaluated fingerprinting algorithms maintain their relative ranking, while the localization error, as expected, usually scales with the size of the environment. Future work includes further variations in parameters, e.g. variations in number of dedicated APs used in each testbed or extending the number of scenarios, in order to get a broader understanding of WiFi fingerprinting-based indoor localization algorithms.

VII. ACKNOWLEDGMENTS

This work has been partially funded by the European Commission (FP7-ICT-FIRE) within the project EVAILOS (grant No. 317989). The authors would also like to thank the EVAILOS project partners for their valuable comments while designing the EVAILOS benchmarking methodology.

REFERENCES