Towards Ubiquitous Indoor Positioning: Comparing Systems across Heterogeneous Datasets

Joaquín Torres-Sospedra*, Ivo Silva†, Lucie Klus‡§, Darwin Quezada-Gaibor‡§, Antonino Crivello¶, Paolo Barsocchi¶, Cristiano Pendão†, Elena Simona Lohan‡, Jari Nurmi‡, and Adriano Moreira§
*UBIK Geospatial Solutions S.L., Castellon, Spain
† Algoritmi Research Center, University of Minho, Guimarães, Portugal
‡Faculty of Information Technology and Communication Sciences, Tampere University, Tampere, Finland
§Institute of New Imaging Technologies, Universitat Jaume I, Castellon, Spain
¶Information Science and Technologies Institute, National Research Council, Pisa, Italy

***Abstract*—The evaluation of Indoor Positioning Systems (IPSs) mostly relies on local deployments in the researchers' or partners' facilities. The complexity of preparing comprehensive experiments, collecting data, and considering multiple scenarios usually limits the evaluation area and, therefore, the assessment of the proposed systems. The requirements and features of controlled experiments cannot be generalized since the use of the same sensors or anchors density cannot be guaranteed. The dawn of datasets is pushing IPS evaluation to a similar level as machine-learning models, where new proposals are evaluated over many heterogeneous datasets. This paper proposes a way to evaluate IPSs in multiple scenarios, that is validated with three use cases. The results prove that the proposed aggregation of the evaluation metric values is a useful tool for high-level comparison of IPSs.**

***Index Terms*—Evaluation; Indoor Positioning Benchmarking**

I. Introduction

In the last decade, IPSs have attracted interest from researchers and industries showing, nowadays, good performance and accuracy in different scenarios and test cases. In literature, especially reading papers from specific conferences and scientific journals, the evolution of these systems is quite clear. Starting from the first works in this field, researchers have shown several techniques and technologies able to accurately estimate a target position, e.g., people or end-users devices. For example, by examining the proceedings of the Indoor Positioning and Indoor Navigation (IPIN) conference, a reader can clearly observe that, up to now, several efforts have been made to create common evaluation frameworks and to set common scenarios to increase the chance of generalizing the results obtained by researchers. With these considerations in mind, it is worth noting the valuable results of the IPIN Competitions [1, 2] in which organizers set several tracks in the same scenarios, offering a common testbed to competitors in order to evaluate their own systems.

Corresponding Author: J. Torres-Sospedra (torres@ubikgs.com)

The authors gratefully acknowledge funding from European Union's Horizon 2020 Research and Innovation programme under the Marie Sklodowska Curie grant agreement No. 813278 (A-WEAR: A network for dynamic wearable applications with privacy constraints, http://www.a-wear.eu/). FCT – Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020 and the PhD fellowship PD/BD/137401/2018. J. Torres-Sospedra acknowledges funding from MICIU (INSIGNIA, PTQ2018-009981)

Organizers of the IPIN Competitions have also proposed an evaluation framework [3], nowadays widely adopted by the research community for evaluating online and offline systems. With the same goal of offering common data and scenarios for evaluation purposes, several researchers have proposed in the last years free and accessible datasets collected in different environments (e.g. hospitals [4], universities [5], malls [6], or factories [7]) and exploiting different positioning technologies.

All those efforts have led to systems and technologies quite stable to hit the market. Nevertheless, the main challenge in this field is to be able to generalize the results and the methods of the systems in heterogeneous environments. In other words, systems are starting to show robust performances when deployed in a specific scenario, but their performances may drop significantly if deployed in a different scenario. In literature, IPSs are shown and tested in real-world scenarios but they are generally developed and tuned to obtain the best performance (e.g., the positioning accuracy) in the considered testing environments. This customization reduces the generalization ability of an IPS to work in any scenario.

The huge availability of public datasets could help researchers to find the best system settings, but typically the IPSs are still only evaluated in one or two scenarios (see Fig. 1). Furthermore, we remark that a shared framework for evaluating over multiple datasets could improve the evaluation process. For example, such a framework can increase the trustiness in sharing scientific results and can enable research reproducibility. We also remark that, a consensus on which features are important during the evaluation process is still missing. In fact, if the overall accuracy is an important metric, the execution time and the computational cost are also fundamental variables for IPSs which have the ambition to overcome the research grade. The main contributions of this paper can be summarized as follows:

- We introduce a novel way to aggregate the evaluation metrics, considering different, heterogeneous, scenarios;
- We propose guidelines and recommendations for guiding researchers in evaluating their positioning systems;
- Through use cases we validate our proposal showing how multiple datasets can improve the IPSs evaluation.

II. Related work

Traditionally, the evaluation of novel Machine Learning (ML) models has included a large setup with multiple diverse datasets, i.e., the evaluation is not limited to just one problem, and it incorporates several datasets covering a heterogeneous set of problems including, for instance, the identification of iris plants, prediction of whether an income exceeds an amount based on census data, the origin of wines, or detection of the presence of a heart disease in a patient, among many others. The traditional datasets can be found in the University of California, Irvine (UCI) machine-learning repository [8].

Bradley investigated the use of the area under the receiver operating characteristic (ROC) curve (AUC) as a performance measure for ML algorithms in [9]. The metric evaluation included a comparison with six ML models and six datasets from UCI. Datasets were diverse and covered issues on post-operative bleeding, breast cancer, diabetes and heart disease.

Yang et al. [10] introduced a survey of face-recognition models where they identified nine datasets valid for training purposes (where each photo contained just one individual) and four datasets for testing purposes (presenting several challenges for face recognition). The datasets were independently collected by Kodak, Harvard University, Yale University, AT&T, and MIT among others. The authors of the survey identified several weaknesses, namely, evaluation using a modest-sized standard test set, "tweaking" the models to get better performance on the test set or even testing on the training set, which is an unacceptable practice in ML. Some of these weaknesses have been often detected in the evaluation of IPSs. For instance, collecting consecutive Received Signal Strength (RSS) fingerprints and directly splitting the collected dataset into training and testing with cross-validation might be considered data leaking. The resulting test set would not be fully independent, driving to over-optimistic accuracy. If the operating system buffering is not taken into account, the same fingerprint vector may end up in the training and test sets. Yang et al. concluded that the fair and effective performance evaluation requires careful design of protocols, scope, and, above all, datasets.

Despite the fact that ML models are usually evaluated with a "moderate" number of datasets as in [11], it is not unusual to find works where the evaluation considers a very large set of diverse datasets. Fernández-Delgado et al. [12] proposed an evaluation with 121 datasets which was later adopted by Zhang et al. [13]. Hynes et al. [14] provided an evaluation over 600 databases hosted on Kaggle. However, a review on ML [15] has shown that most of recent works are evaluated with one or two datasets, and that an evaluation with three or more datasets is less frequently encountered in the current literature. The new trends on deep learning applied to particular problems and the existence of extremely large datasets (with millions of samples) have driven the ML algorithms to computationally demanding evaluation procedures. However, in localization domain, an evaluation considering many "traditional" moderate-sized datasets is still possible [16].

Olson et al. [17] provided a comparison of a few ML models using 165 real-world curated datasets from the Penn Machine Learning Benchmark (PMLB) suite, which currently has has 299 datasets (April 2021). Several useful ways to summarize the full results as images were also provided. Other ML tools – such as WEKA, *scikit-learn* , TensorFlow or Keras– are easing the integration of ML models in real-world implementations.

However, the level of evaluation carried out in the ML domain is often not reached in the indoor-positioning area, which usually relies on the evaluation of a single setup with very controlled conditions. Collecting data for wireless indoor positioning is a time-consuming and demanding procedure. However, there are many attempts to provide public datasets in this domain. Fruit of those datasets, Saccomanno et al. [16] provided a comprehensive study where the relation between Wi-Fi fingerprints and the spatial knowledge was explored for indoor positioning using multiple datasets, which extended the setup previously provided by Torres-Sospedra et al. in [18, 19].

Although there are several datasets and database repositories [20, 21] available for IPS evaluation, most research papers still use their own closed setups or datasets. The 79 accepted papers of the IPIN 2019 conference were analysed to have a better picture of the current trends in evaluating IPS. The results of the review are shown in Fig. 1, where it can be seen that most of the works with empirical evaluation only included 1 or 2 scenarios, and only two papers included 3 [22] and 4 [23] scenarios respectively. These data lead to remark the importance of finding a common benchmark for evaluation purpose. The efforts in terms of standardization are increasing and, in this context, guidelines and strategies for comparing IPS in several and heterogeneous scenarios represent a step forward in this research field. In fact, in order to understand which technologies and techniques are more able to fit in different environments, we should promote the adoption of common datasets, and as future works we should probably try to standardizing them, to promote reliable and robust systems.

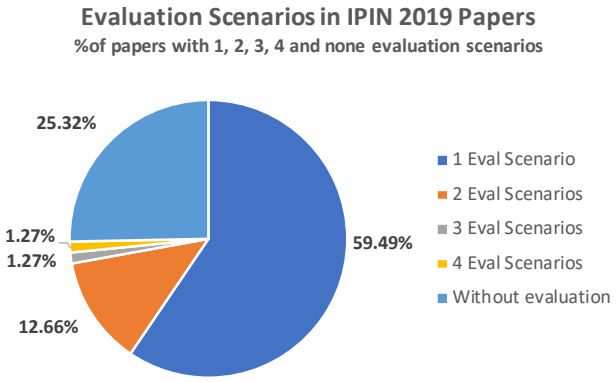

Fig. 1. Analysis of the evaluation of the regular papers presented in the IPIN 2019 Conference

In this paper, we would like to mimic the evaluation procedures in ML, this time applied to IPSs, and show the importance of evaluating IPSs with multiple diverse datasets with the proposed aggregated metrics and visualization tools.

III. Materials and Methods

This section focuses on the proposed methodology to aggregate metrics and to how perform visual comparisons.

A. Aggregating evaluation metrics

Let's suppose that we follow the Black-box testing approach suggested in the ISO18305 [24] and discussed in [25]. Given an IPS, its evaluation metric can be represented by:

$$\mathcal{M}_{method}^{scenario,trial} \tag{1}$$

where $\mathcal{M}$ represents the evaluation metric (e.g., mean positioning error, third quartile error, root mean squared error, floor hit rate, execution time, among many others), $method$ identifies the evaluated method, $scenario$ corresponds to the evaluation scenario and $trial$ corresponds to the trial number (or execution run) for that scenario. The best value for a metric depends on its nature. For instance, developers want to provide IPS with low positioning error and high floor detection rate.

It is worth noting that in a dataset-based evaluation, the scenario corresponds to the dataset itself, whereas the trial corresponds to the execution run. In an on-line evaluation, without datasets, the scenario can be considered the combination of the evaluation area and the positioning infrastructure.

In the simplest evaluation, with a single metric, scenario and trial, several methods can be directly compared, i.e. the method reporting the best metric value can be considered the best solution. However, real-world evaluation may include multiple runs or trials under the same scenario as the IPS might be affected by environmental conditions or a random initialization. For instance, in the IPIN annual competition, the participants are able to provide multiple runs being the trial with providing the best accuracy the used for ranking. In other domains, such as machine learning where some models depend on random initialization, the average over the multiple runs is applied to obtain the final value for the metric. In this paper, we aggregate the results for multiple trials as the average among all the N_{trials} number of trials.

$$\bar{\mathcal{M}}_{method}^{scenario} = \frac{\sum_{t=1}^{N_{trials}} \mathcal{M}_{method}^{scenario,t}}{N_{trials}} \tag{2}$$

As the metric values usually depend on the scenario, we propose to normalize the metric to a base line becoming the unitless metric. That normalization should be done with respect to a simple method or configuration. In RSS-based fingerprinting models, the baseline could be the 1-NN algorithm.

$$\hat{\mathcal{M}}_{method}^{scenario} = \frac{\bar{\mathcal{M}}_{method}^{scenario}}{\mathcal{M}_{baseline}^{scenario}} \tag{3}$$

In addition, a comprehensive evaluation should include different scenarios covering multiple cases, since an indoor positioning solution may behave differently in two different scenarios. To integrate different scenarios, we propose to report the aggregated-values average and the standard deviation of the baseline-normalized values for all scenarios as follows:

$$\tilde{\mathcal{M}}_{method} = \frac{\sum_{scenario=1}^{N_{scenarios}} \hat{\mathcal{M}}_{method}^{scenario}}{N_{scenarios}} \tag{4}$$

The average of the baseline-normalized values is providing the general behaviour of the method considering multiple scenarios, whereas the standard deviation reflects the variability of the metric along all the scenarios considered. When evaluating an IPS, we target those methods providing the best averaged value with the lowest possible deviation. It is worth noting that here we use the term "best" on purpose as there are metrics, such as the ones based on the positioning error, where the averaged values should be as lowest as possible. On the other hand in metrics such as the floor identification rate (percentage of correctly identified floor number), the averaged values must be as high as possible.

As the on-line evaluation of IPSs is very demanding, we will integrate the proposed approach to aggregate the results off-line. Thus, the IPSs will be evaluated with pre-recorded datasets with independent dedicated training and evaluation sets. Using the same dataset over the different trials should not affect the evaluation metrics based on the positioning error, as the data used is the same in the multiple runs. However, if the method employed relied on a kind of random initialization (such as a Neural Network), then the positioning error should vary from run to run.

B. Comparing two different metrics over multiple datasets

Evaluation becomes more complex when several targets must be accomplished. For instance, one could aim simultaneously at providing the lowest possible 3D positioning error and the lowest execution time. With the proposed way to aggregate a metric over different scenarios and trials, the simplest option is to provide the information as a table with as many columns as metrics considered. If the number of metrics is two, it can be complemented with a scatter plot showing the average of the baseline-normalized values for the two metrics.

If one would like to go a step further and show the results for each dataset with the two metrics, we propose a novel graphical representation (we call it a GMMS plot), which provides four dimensions in a single plot. The x-axis corresponds to the scenario (or dataset), the y-axis to the method and each plotted element is a colored ellipse whose color (green range, white and red range) indicates one evaluation metric and the shape (horizontal, circled, vertical) stands for the other metric. Fig. 2 shows an example on how elements are displayed in a GMMS plot according to the aggregated mean values of positioning error $\tilde{\epsilon_{3D}}$ and dataset execution time, $\tilde{\tau_{DB}}$. With respect to the baseline, the more red the element is the worse (higher value) $\tilde{\tau_{DB}}$ is, in the same way the more vertical the element is the higher $\tilde{\epsilon_{3D}}$ is. On the contrary, the greener the element is, the lower $\tilde{\tau_{DB}}$ is and, in the same way, the more horizontal the element is, the lower $\tilde{\epsilon_{3D}}$ is. For the baseline values, we use a white circle.

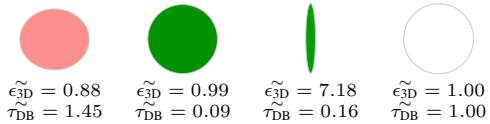

Fig. 2. Example of elements in the GMMS plot

IV. USE CASES

A. Analysis on the parameters of the k-NN algorithm

As the first use case, we provide a general analysis of the distance metrics for the k-Nearest Neighbour (NN) algorithm used in fingerprinting. The intention is not to analyse the algorithm, but to show the potential of the proposed aggregation metrics to perform a more general comparison. We have considered 16 public as in [16, 18, 19] and the experiments were run in a computer with Intel Core i7-8700 CPU and Octave 4.0.3. Moreover, we consider two evaluation metrics the mean positioning error, ϵ_{3D}, and the dataset execution time, τ_{DB}. Due to the lack of space, we only show the aggregated metrics and comment on particular results.

First, the analysis was run using the k-NN algorithm with the following configuration: $k = 1$, positive data representation [26] and the city block distance as distance/similarity metric. This plain version of the k-NN can be considered our baseline for further comparisons, whose full results are reported in Table I. Please note that the metrics $\bar{\epsilon_{3D}}$ and $\bar{\tau_{DB}}$ provides the averaged values over 10 runs, whereas the metrics $\hat{\epsilon_{3D}}$ and $\hat{\tau_{DB}}$ are normalized to the baseline. The aggregated metrics $\tilde{\epsilon_{3D}}$ and $\tilde{\tau_{DB}}$ provides the average and standard dev. of the normalized values across the 16 datasets.

TABLE I
FULL RESULTS OF 1-NN, CITY BLOCK DISTANCE AND POSITIVE DATA REP.

	Absolute values		Norm. values	
Dataset	$\bar{\epsilon_{3D}}$ (m)	$\bar{\tau_{DB}}$ (s)	$\hat{\epsilon_{3D}}$	$\hat{\tau_{DB}}$
DSI 1	4.95	12.21	1	1
DSI 2	4.95	5.15	1	1
LIB 1	3.02	46.19	1	1
LIB 2	4.18	46.39	1	1
MAN 1	2.82	155.46	1	1
MAN 2	2.47	14.26	1	1
SIM	3.24	252.00	1	1
TUT 1	9.59	18.93	1	1
TUT 2	14.37	2.73	1	1
TUT 3	9.59	79.73	1	1
TUT 4	6.36	79.88	1	1
TUT 5	6.92	11.88	1	1
TUT 6	1.94	620.72	1	1
TUT 7	2.69	511.70	1	1
UJI 1	10.81	599.04	1	1
UJI 2	8.05	2924.69	1	1
			$\tilde{\epsilon_{3D}}$ mean(std)	$\tilde{\tau_{DB}}$ mean(std)
	Plain k-NN (baseline)		1.00 (0.00)	1.00 (0.00)

On the one hand, it can be clearly observed that the execution time of the entire evaluation dataset (τ_{DB}) highly depends on the dataset, as k-NN computational cost depends on the number of training and evaluation samples. On the other hand, the mean positioning error varies, ranging from almost 2 m (TUT 6), to more than 14 m (TUT 2). This variability on the timing and accuracy measurements might make a direct comparison difficult. For example, a reduction of 50 cm in the positioning error is more significant in dataset TUT 6 than in TUT 2. Similarly, a reduction of 2 s in the execution time is more significant in dataset TUT 2 than in TUT 6.

Second, the analysis on the distance function used to compare two fingerprints is shown in Table II. We provide the aggregated positioning error $\tilde{\epsilon_{3D}}$ and the execution time $\tilde{\tau_{DB}}$ of all the alternatives. For both metrics, we provide the average and the standard deviation of the baseline-normalized values over the 16 datasets. The distance metrics were evaluated keeping the other two parameters of the baseline configuration unaltered (positive data representation and $k = 1$).

TABLE II
COMPARISON OF THE AGGREGATED VALUES FOR THE POSITIONING ERROR AND EXECUTION TIME FOR 1-NN WITH 14 DISTANCE METRICS.

Distance	$\tilde{\epsilon_{3D}}$	$\tilde{\tau_{DB}}$	Distance	$\tilde{\epsilon_{3D}}$	$\tilde{\tau_{DB}}$
$\text{Kulczynski}_\text{d}$	0.90 (0.12)	1.84 (0.02)	City Block	1.00 (0.00)	1.00 (0.00)
$\text{Kulczynski}_\text{s}$	0.90 (0.12)	1.86 (0.01)	LGD	1.00 (0.22)	2.09 (0.25)
Motyka	0.90 (0.12)	1.18 (0.01)	PLGD10	0.91 (0.15)	3.15 (0.32)
Ruzicka	0.90 (0.12)	1.30 (0.01)	PLGD40	0.95 (0.19)	3.15 (0.32)
Soergel	0.90 (0.12)	1.29 (0.01)	Euclidean	0.99 (0.05)	1.02 (0.01)
Sørensen	0.90 (0.12)	1.16 (0.01)	Neyman	1.28 (0.36)	1.59 (0.04)
Tanimoto	0.90 (0.12)	1.48 (0.01)	Euclidean^2	0.99 (0.05)	0.92 (0.01)

The left side of Table II shows the results on some distances that are equivalent between them in terms of sorting the reference samples by distance to the operational sample, and therefore they provide the same positioning errors (10 % lower than in the baseline for all of them). However, they differ in terms of computational costs, with an increase in costs ranging from 16 % to 86 % on average. Among these equivalent metrics, the Sørensen distance is the one reporting the best computational costs (only 16 % higher with respect to the baseline). In general, the Sørensen distance is reporting lower positioning errors than the Euclidean distance (baseline distance metrics) in most of the datasets, providing a mean positioning error 25 % lower than the baseline for TUT 1.

The right side of Table II shows the results on the remaining metrics. Although the Euclidean distance and city block are not equivalent, they are providing similar general results in terms of averaged normalized positioning error when considering all the datasets (0.99 and 1.0 respectively) and computational costs (1.02 and 1.0 respectively). Despite the former is performing one square per Access Point (AP) and one square root operations and the later computes one absolute value per AP, their computational times are almost the same in all datasets ($\tau_{DB} = 1 \pm 0.05$). Despite these similarities in the averaged case, their performance clearly depends on the dataset (for UJI 1 the Euclidean distance is providing an error 12 % lower than the city block distance, but it is 7 % higher for TUT 6). The Squared Euclidean (Euclidean^2 in Table II) is equivalent to the Euclidean distance in terms of ranking samples by distance but it has a lower computational cost (1.02 vs 0.92), since the square root operation is not needed to obtain an equivalent samples ranking. The three Log Gaussian-based distances (LGD, PLGD10 and PLGD40) have attached a significant increase of the computation time, but in some cases they provide a great improvement on the positioning error (e.g., TUT 1 and PLGD40, where the error has been reduced a 45 % with respect to the baseline).

In the election of the best distance metric, some concerns about the computational costs might raise. For instance, PLDG40 is better than Sørensen for dataset SIM, their difference in the normalized positioning error is just 2 % (around 7 cm if we consider the absolute positioning errors) but the normalized computational costs are very different, 3.15 and 1.16 respectively. This means that to reach similar averaged accuracy, the process to estimate the position takes more than two times with PLGD40 than with Sørensen distance and the gain in accuracy might be considered marginal. In our opinion, the election of the best alternative should balance both metrics. In case of similar positioning error, we should select the one that is computationally efficient (green computing).

Finally, this analysis has shown that the distance/similarity function does not only impact the positioning accuracy but also the computational burden. The average of the baseline-normalized values for all datasets provides the general behavior. The standard deviation identifies where there exists a huge dependency on the dataset and dataset-based analysis is needed to select the optimal value for the parameter.

B. Comparison on clustering models for Wi-Fi fingerprinting

The second use case corresponds to the comparison of clustering models to make faster the estimation of the indoor position using k-NN algorithm. It is well known that k-NN does not require a training phase but, in contrast, it is inefficient as it needs to compute the distance/similarity function between the operational fingerprint and all the reference fingerprints in the radio map.

One alternative to alleviate the computational burden is to apply clustering models to the radio map, by generating clusters that group fingerprints with similar features. Then, in the operational phase one has to first search for the most similar group (cluster) and then compute the distance function to all the reference fingerprints falling into that cluster.

For the experiments, we have used the 16 datasets from the first use case and the experiments were carried out on the same desktop computer. The configuration of the k-NN estimator for all the methods corresponds to the baseline previously used with k equal to 1, the positive data representation and the city block distance as similarity measure for fingerprint comparison. We consider two evaluation metrics: the mean positioning error, ϵ_{3D} and the dataset execution time, τ_{DB} to assess the performance of the clustering models.

Table III and Fig. 3 show the results of some well-known clustering models (k-Means, k-Medoids, Fuzzy c-Means, Affinity Propagation, DBSCAN, HDBSCAN and Model-based) in the literature. k-Means, k-Medoids and Fuzzy c-Means require the number of clusters as an input parameter. For them, we have tested three values: 25; the square root of the number of reference fingerprints in the radio map ($rfp1$); and the number of reference fingerprints in the radio map divided by 25 ($rfp2$). For DBSCAN-based methods, we used the optimal values for the minimum number of points and the distance used to locate the points in the neighbourhood.

TABLE III
RESULTS REPORTED BY THE SELECTED CLUSTERING METHODS

method	params	$\widetilde{\epsilon_{3D}}$ mean(std)	$\widetilde{\tau_{DB}}$ mean(std)
plain 1-NN	–	1.00 (0.00)	1.00 (0.00)
k-means	$k = 0025$	1.03 (0.03)	0.10 (0.03)
k-means	$k = rfp1$	1.05 (0.05)	0.07 (0.04)
k-means	$k = rfp2$	1.06 (0.06)	0.08 (0.04)
k-medoids	$k = 0025$	1.06 (0.05)	0.11 (0.04)
k-medoids	$k = rfp1$	1.08 (0.07)	0.08 (0.05)
k-medoids	$k = rfp2$	1.09 (0.07)	0.08 (0.04)
c-means	$c = 0025$	4.92 (8.92)	0.20 (0.17)
c-means	$c = rfp1$	4.16 (5.19)	0.19 (0.13)
c-means	$c = rfp2$	1.60 (0.84)	0.16 (0.15)
Affinity Propagation	–	1.10 (0.08)	0.09 (0.04)
DBSCAN	best params	2.01 (1.35)	0.12 (0.11)
HDBSCAN	best params	2.18 (3.37)	0.31 (0.31)
Model Based	–	3.92 (4.70)	0.22 (0.30)

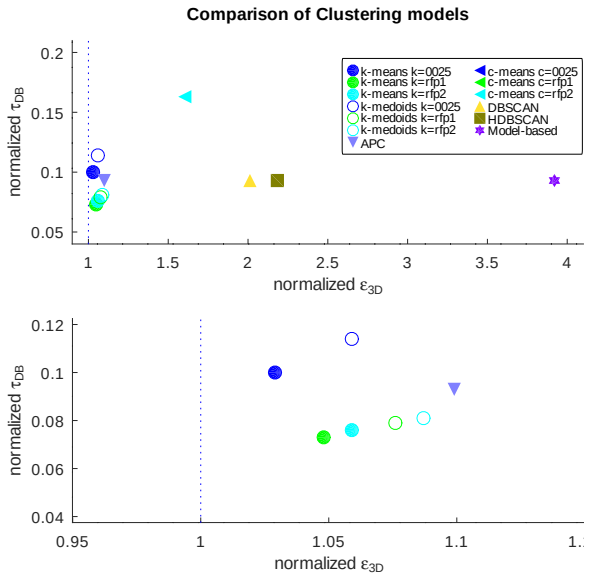

Fig. 3. Visualization of the aggregated results reported by the clustering methods (top figure) and zoom to the best results (bottom figure)

The results reported in the table and figure show that all the clustering models reduce the computational time of fingerprinting. However, the aggregated computational time is high for HDBSCAN and Model Based as, in a few datasets, they failed at creating the clusters. Regarding the aggregated accuracy, we should discard DBSCAN, HDBSCAN, Model-based and c-Means as the aggregated positioning error is so high. The large variability might also indicate that those models do not work well in certain circumstances. In fact, according to the GMMS plot in Fig. 4, they do not work for UJI 1 and some TUT datasets (vertical ellipsoids).

In the bottom part of Fig. 3, we can see that k-Means, k-Medoids and Affinity Propagation Clustering are very similar, being the k-means with $k = rfp1$ presenting the best trade-off between positioning error and execution time. The aggregated metrics have proven to be useful to compare method as a full analysis would be not possible. As an example, we provide the full results for k-means with $k = rfp1$ in Table IV.

TABLE IV
FULL RESULTS FOR k-MEANS WITH $k = rfp1$

Dataset	Mean Positioning Error – ϵ^i_{3D} (m), ϵ^{-}_{3D} (m) and $\epsilon^{\wedge}_{3D}$ (unitless)												Execution Time – τ^i_{DB} (s), τ^{-}_{DB} (s) and $\tau^{\wedge}_{DB}$ (unitless)											
	ϵ^1_{3D}	ϵ^2_{3D}	ϵ^3_{3D}	ϵ^4_{3D}	ϵ^5_{3D}	ϵ^6_{3D}	ϵ^7_{3D}	ϵ^8_{3D}	ϵ^9_{3D}	ϵ^{10}_{3D}	ϵ^{-}_{3D}	$\epsilon^{\wedge}_{3D}$	τ^1_{DB}	τ^2_{DB}	τ^3_{DB}	τ^4_{DB}	τ^5_{DB}	τ^6_{DB}	τ^7_{DB}	τ^8_{DB}	τ^9_{DB}	τ^{10}_{DB}	τ^{-}_{DB}	$\tau^{\wedge}_{DB}$
DSI1	4.93	4.97	5.22	5.21	4.99	5.40	5.30	5.29	5.08	4.85	5.13	1.04	0.79	0.86	0.97	0.86	0.79	0.81	0.81	1.01	0.90	0.91	0.87	0.07
DSI2	4.94	5.13	5.01	5.25	4.72	4.88	5.10	5.75	4.92	4.78	5.05	1.02	0.59	0.52	0.53	0.54	0.53	0.53	0.60	0.52	0.62	0.58	0.56	0.11
LIB1	3.10	3.16	3.14	3.10	3.13	3.11	3.11	3.12	3.12	3.19	3.13	1.04	4.32	4.77	4.21	4.17	4.42	4.17	4.40	4.50	4.55	4.33	4.38	0.09
LIB2	4.29	4.18	4.26	4.54	4.07	4.27	4.22	4.19	4.16	4.42	4.26	1.02	4.79	5.64	4.54	4.47	5.45	4.62	4.98	4.92	5.67	4.81	4.99	0.11
MAN1	2.85	2.89	2.82	2.88	2.95	2.84	2.97	2.84	2.94	2.82	2.88	1.02	2.96	3.08	2.92	2.92	2.82	2.89	2.95	3.02	2.98	2.92	2.95	0.02
MAN2	2.62	2.46	2.48	2.56	2.45	2.43	2.40	2.60	2.35	2.45	2.48	1.01	0.96	0.93	0.92	1.04	0.98	0.94	0.88	0.92	0.98	1.02	0.96	0.07
SIM	3.27	3.28	3.36	3.33	3.28	3.35	3.31	3.27	3.37	3.35	3.32	1.03	5.00	4.95	4.93	5.02	4.88	4.88	4.98	4.94	4.89	4.85	4.93	0.02
TUT1	10.06	9.43	9.76	10.03	8.99	10.12	10.77	9.79	9.37	10.36	9.87	1.03	1.25	1.15	1.11	1.11	1.06	1.15	1.39	1.17	1.12	1.19	1.17	0.06
TUT2	13.84	13.39	16.02	12.42	13.98	14.19	13.33	14.35	13.91	16.83	14.22	0.99	0.29	0.33	0.29	0.29	0.31	0.30	0.28	0.35	0.30	0.30	0.30	0.11
TUT3	9.96	10.02	10.02	10.10	9.92	9.86	10.14	9.88	10.05	9.94	9.99	1.04	16.99	11.62	13.30	10.79	13.69	13.29	12.20	13.28	11.52	13.25	12.99	0.16
TUT4	6.62	6.74	6.64	6.67	6.74	6.48	6.54	6.60	6.59	6.69	6.63	1.04	5.12	5.31	5.12	4.84	8.17	4.82	5.50	4.88	6.08	4.94	5.48	0.07
TUT5	7.74	7.29	7.14	7.13	7.12	7.50	7.51	7.40	7.37	7.10	7.33	1.06	1.36	1.30	1.49	1.53	1.39	1.37	1.46	1.63	1.35	1.62	1.45	0.12
TUT6	2.25	2.14	2.20	2.11	2.17	2.19	2.21	2.27	2.20	2.13	2.19	1.13	37.47	31.15	35.34	45.38	40.16	33.18	40.99	39.89	33.41	41.70	37.87	0.06
TUT7	2.91	2.84	2.92	2.87	2.92	2.90	2.88	2.87	2.84	2.93	2.89	1.07	30.48	42.74	29.63	27.72	33.01	31.98	31.57	34.87	29.53	35.86	32.74	0.06
UJI1	12.76	12.49	13.01	13.10	12.28	12.78	12.72	12.85	13.06	13.23	12.83	1.19	11.75	15.42	12.34	12.76	12.40	11.61	13.61	13.38	10.52	11.95	12.57	0.02
UJI2	8.72	8.36	8.89	8.54	8.43	8.40	8.51	8.36	8.53	8.61	8.54	1.06	43.39	50.69	48.61	44.23	44.27	49.47	44.85	44.20	49.26	47.02	46.60	0.02
											$\tilde{\epsilon}_{3D}$	mean											$\tilde{\tau}_{3D}$	mean
												std												std
												(0.05)												(0.04)

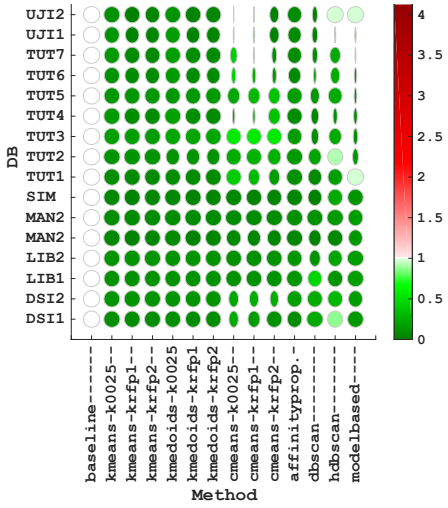

Fig. 4. GMMS representation for the results provided by the clustering methods for the 16 datasets. The *baseline* is the plain 1-NN method.

Despite the fact that a better analysis can be performed with the full results (see Table IV), one can focus on them after filtering out those methods that do not provide good general results. After selecting k-means with $k = rfp1$ as best choice, we can see that the model provides relative good results except for dataset UJI 1. In general, the computational time reduction is significant in all datasets, especially in those that have large radio maps (UJI 1, UJI 2, SIM, and MAN 1). The weakest point of k-Means, even providing the best general results, is that its accuracy and execution time depend on the partition done over the radio map. The variability in the error between runs is evident in Table IV, where the superscript in the evaluation metrics stand for the trial/run. k-means with $k = rfp1$ has similar general accuracy that the baseline in the best run, whereas it is similar to affinity propagation (which provides deterministic clustering) in the worst.

C. Data Compression

The third use case, in which we show the benefits of the aggregated metrics, is based on studying the impact of the database compression using Adaptive k-Means (AkM) algorithm [27] on its positioning capabilities. In this use case, we would like to apply the aggregated metrics to first determine the optimal parametrisation of AkM and then to compare to the plain 1-NN.

For the first part, to determine the optimal parameters, we cannot use the same baseline method for results normalization, as some metrics are dependent on the clustering approach selected. In particular, the studied metrics are Mean Square Error (MSE) after Stage 1 and Stage 2 of the AkM compression (MSE_{S1} and MSE_{S2}) , Normalized Mean Positioning Error ($\tilde{\epsilon}_{3D}$), and achieved Compression Ratio (CR).

The metrics MSE_{S1} and MSE_{S2} correspond to the mean squared difference between the original RSSI values and the RSSI values after compression using clustering. We analyze the AkM algorithm by modifying the number of clusters K. The algorithm is initiated by clustering all train dataset samples' RSS values separately, thus first creating a one-dimensional array of all entries of the train RSS samples. The algorithm further clusters the array into K clusters, each of them defined by its centroid coordinate (a single number). The MSE_{S1} is then calculated as MSE between the original test RSS data and its reconstruction after clustering using the previously obtained centroids. The algorithm then adapts the centroid coordinates by including the test RSS data, which results in shifting the centroid coordinates as described in [27]. MSE_{S2} is then calculated as MSE between the original test RSS data and its reconstruction using the adapted centroid coordinates. The algorithm then applies the k-NN regression with k = 1 on the clustered train and test datasets (where the original RSS values were replaced by the corresponding centroid coordinates) and obtains the positioning predictions for the test samples.

The CR is then calculated as number of bits of the original data divided by the number of bits of the compressed data. We assume that the uncompressed integer-valued RSS measurements are saved in 7-bit format (allowing 128 unique RSS values) and that $ceil(log_2(K))$ unique values can be saved with K bits, where ceil() rounds up to the next higher integer.

Therefore, for comparing the different setups of AkM, we have chosen the simplest version with $K = 2$ (for k from k-Means) as baseline and, then, we normalize the four metrics to it. The aggregated results on all 16 considered datasets are shown in Table V and are achieved by averaging over 10 repetitions of the algorithm for each dataset per each K setting. The considered K values for AkM clustering are 2, 4, 7, 15, 25 and 35. As the number of clusters increases, the aggregated CR decreases accordingly, depending on the number of required bits to compress each value. The aggregated MSE_{S1} and MSE_{S2} parameters also decrease with increasing K parameter. This is a natural result of decreasing rounding error during clustering (larger number of clusters leads to smaller distance each sample is shifted during clustering). Finally, the aggregated positioning error $\tilde{\epsilon}_{3D}$ also decreases.

TABLE V
RESULTS FOR SETTING THE BEST OVERALL PARAMETERS FOR AKM

K	$\tilde{MSE}_{S1}$	$\tilde{MSE}_{S2}$	$\tilde{\epsilon}_{3D}$	$\tilde{CR}$	$\tilde{\mathcal{F}}$
2	1.000 (0.00)	1.000 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00
4	0.163 (0.04)	0.164 (0.04)	0.84 (0.10)	0.50 (0.00)	0.76
7	0.050 (0.02)	0.051 (0.02)	0.81 (0.11)	0.33 (0.00)	0.73
15	0.010 (0.00)	0.010 (0.00)	0.79 (0.12)	0.25 (0.00)	**0.72**
25	0.003 (0.00)	0.003 (0.00)	0.79 (0.12)	0.20 (0.00)	0.72
35	0.001 (0.00)	0.001 (0.00)	0.79 (0.12)	0.17 (0.00)	0.73

However, given the results reported in the table, it is not that easy to retrieve a winning setup, as larger K values (for k-Means) lead to better results but lower compression. After some discussion, we decided for that particular problem an additional aggregation in the way:

$$\tilde{\mathcal{F}} = 0.05 \cdot \tilde{MSE}_{S1} + 0.05 \cdot \tilde{MSE}_{S2} + 0.9 \cdot (\tilde{\epsilon}_{3D})^2 + 0.2 \cdot (1 - \tilde{CR})$$

where the aggregated positioning error has more weight than the Compression Ratio (CR) and the two metrics based on the MSE, becoming $K = 15$ the best configuration for AkM.

Table VI compares the plain 1-NN with Adaptive k-Means (AkM) and $k = 15$, using 1-NN as baseline. The results show that the selection of parameters led AkM to provide similar accuracy as 1-NN with a more efficient RSSI representation.

TABLE VI
COMPARISON OF PLAIN 1-NN WITH AKM

method	$\tilde{\epsilon}_{3D}$ mean(std)	$\tilde{CR}$ mean(std)
Plain 1-NN.	1.00 (0.00)	1.00 (0.00)
AkM ($k = 15$)	1.00 (0.03)	1.75 (0.00)

TABLE VII
FULL RESULTS OF AKM.

	Absolute values		Norm. values	
Dataset	$\bar{\epsilon}_{3D}$ (m)	$\bar{CR}$ (s)	$\hat{\epsilon}_{3D}$	$\hat{CR}$
DSI 1	4.88	1.75	0.99	1.75
DSI 2	5.21	1.75	1.02	1.75
LIB 1	3.04	1.75	1.01	1.75
LIB 2	4.22	1.75	1.01	1.75
MAN 1	2.88	1.75	1.02	1.75
MAN 2	2.39	1.75	0.97	1.75
SIM	3.60	1.75	1.10	1.75
TUT 1	9.75	1.75	1.02	1.75
TUT 2	14.25	1.75	0.99	1.75
TUT 3	9.55	1.75	1.00	1.75
TUT 4	6.35	1.75	1.00	1.75
TUT 5	6.98	1.75	1.01	1.75
TUT 6	1.98	1.75	1.02	1.75
TUT 7	2.71	1.75	1.01	1.75
UJI 1	10.21	1.75	0.94	1.75
UJI 2	7.92	1.75	0.98	1.75

V. DISCUSSION & CONCLUSIONS

In this paper, the importance of evaluating IPSs in multiple scenarios was discussed. This is an essential step for the characterization of the true performance of an IPS, and by consequence, the fair comparison with other solutions. However, due to the complexity involved in preparing experiments in multiple scenarios, dataset publishing by the research community is of utmost importance. Moreover, considering different performance metrics over multiple scenarios, an aggregation of the evaluation metrics is proposed to enable high-level comparison of IPSs.

When dealing with an evaluation that involves multiple metrics, an interesting approach to explore is their combination into a single metric. In such a case, we suggest to apply the following weighted combination of the aggregated metrics:

$$\tilde{\mathcal{F}}_{method} = \omega_{\mathcal{M}} \cdot \tilde{\mathcal{M}}_{method} + \ldots + \omega_{\mathcal{Q}} \cdot \tilde{\mathcal{Q}}_{method} \quad (5)$$

where the weight values (ω) can be user-defined. A weighted combination allows taking into consideration the requirements of a particular use case or IPS deployment, being possible to define how important each of the performance metrics is for the overall evaluation. An IPS may be the best for one use case but not for another with different requirements.

The proposed aggregation of evaluation metrics simplifies complex comparisons between IPSs when considering many parameters. In addition, it also simplifies the process of selecting the best IPS for a specific scenario or deployment.

The use cases we described in this work give us the possibility of validating our proposal to show how multiple datasets can improve the IPSs evaluation. This work highlights also the need of a common and shared graphical representation able to give immediately the impression of what kind of methods are better in all the considered datasets. In addition, we were able to provide results on three different independent experiments in an 8-page paper, demonstrating also the power of the aggregated metrics to summarize general results in a compact table or figure.

However, the evaluation of an IPS is not easy when multiple metrics need to be considered. The combination of several metrics such as accuracy, installation complexity, user acceptance, availability and integrability as done in [28] would need further discussion. Nevertheless, we suggest a new level of aggregation based on a user-defined weighted combination.

Finally, we have found two major issues in the literature. The first one is the lack of guidelines to prepare, collect and publish datasets for indoor positioning. That could be a reason why the community is not adopting datasets in their evaluation as they may not be interoperable with their research. The second one is that most datasets are RSSI-based, the community needs datasets covering other positioning technologies. As the proposed aggregation is agnostic to the positioning technology.

ACKNOWLEDGMENT

We would like to thank Germán Martín Mendoza-Silva and Philipp Richter for their invaluable advice, which encouraged us to prepare the current work presented in this paper.

CREDIT STATEMENT

J. Torres-Sospedra: Conceptualization Ideas; Methodology Development; SW Programming, Validation; Formal analysis; Investigation; Resources; Data Curation; Writing (Original Draft, Review & Editing); and Supervision;
I. Silva: Discussion and Writing (Review & Editing).
L. Klus: Methodology Development; Software Programming, Validation; Investigation; Writing (Review & Editing);
D. Quezada-Gaibor: Software Programming, Validation; Investigation; Writing (Review & Editing);
A. Crivello: Methodology Development; Formal analysis; Investigation and Writing (Original Draft, Review & Editing).
P. Barsocchi: Conceptualization Ideas; Investigation; Writing (Original Draft, Review & Editing); Supervision.
C. Pendão: Conceptualization Ideas and Writing (Original Draft, Review & Editing);
E. S. Lohan: Methodology Development; Resources; Writing (Review & Editing); and Supervision;
J. Nurmi: Conceptualization Ideas; Resources; Writing (Review & Editing).
A. Moreira: Conceptualization Ideas, Methodology Development and Supervision

REFERENCES

[1] V. Renaudin, M. Ortiz, J. Perul, *et al.*, "Evaluating indoor positioning systems in a shopping mall: The lessons learned from the ipin 2018 competition," *IEEE Access*, vol. 7, pp. 148 594–148 628, 2019.
[2] F. Potortì, S. Park, A. Crivello, *et al.*, "The ipin 2019 indoor localisation competition—description and results," *IEEE Access*, vol. 8, pp. 206 674–206 718, 2020.
[3] F. Potortì, S. Park, A. R. Jimenez Ruiz, *et al.*, "Comparing the performance of indoor localization systems through the evaal framework," *Sensors*, vol. 17, no. 10, p. 2327, 2017.
[4] Z. Iqbal, D. Luo, P. Henry, *et al.*, "Accurate real time localization tracking in a clinical environment using bluetooth low energy and deep learning," *PLOS ONE*, vol. 13, no. 10, pp. 1–13, Oct. 2018.
[5] G. M. Mendoza-Silva, P. Richter, J. Torres-Sospedra, *et al.*, *Long-Term Wi-Fi fingerprinting dataset and supporting material*, [Available Online] https://doi.org/10.5281/zenodo.3748719, Zenodo, 2020.
[6] J. A. Lopez Pastor, A. Ruiz Ruiz, A. J. García Sänchez, *et al.*, *Wi-Fi RSSI fingerprint dataset from two malls with validation routes in a shop-level for indoor positioning*, version 1, Zenodo, Mar. 2020.
[7] A. Moreira, M. J. Nicolau, I. Silva, *et al.*, *Wi-Fi Fingerprinting dataset with multiple simultaneous interfaces*, version 1.0, [Available On-line] https://doi.org/10.5281/zenodo.3342526, Zenodo, Sep. 2019.
[8] D. Dua and C. Graff, *UCI machine learning repository*, 2017.
[9] A. P. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," *Pattern Recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.
[10] M.-H. Yang, D. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34–58, 2002.
[11] S.-W. Lin, K.-C. Ying, S.-C. Chen, *et al.*, "Particle swarm optimization for parameter determination and feature selection of support vector machines," *Expert Systems with Applications*, vol. 35, no. 4, pp. 1817–1824, 2008.
[12] M. Fernández-Delgado, E. Cernadas, S. Barro, *et al.*, "Do we need hundreds of classifiers to solve real world classification problems?" *Journal of Machine Learning Research*, vol. 15, no. 90, 2014.
[13] L. Zhang and P. Suganthan, "A comprehensive evaluation of random vector functional link networks," *Information Sciences*, vol. 367-368, pp. 1094–1105, 2016.
[14] N. Hynes, D. Sculley, and M. Terry, "The data linter: Lightweight automated sanity checking for ml data sets," 2017.
[15] J. M. Zhang, M. Harman, L. Ma, *et al.*, "Machine learning testing: Survey, landscapes and horizons," *IEEE Transactions on Software Engineering*, 2020.
[16] N. Saccomanno, A. Brunello, and A. Montanari, "What you sense is not where you are: On the relationships between fingerprints and spatial knowledge in indoor positioning," *IEEE Sensors Journal*, 2021.
[17] R. S. Olson, W. La Cava, P. Orzechowski, *et al.*, "PMLB: a large benchmark suite for machine learning evaluation and comparison," *BioData Mining*, vol. 10, no. 1, p. 36, 2017.
[18] J. Torres-Sospedra, D. Quezada-Gaibor, G. M. Mendoza-Silva, *et al.*, "New cluster selection and fine-grained search for k-means clustering and wi-fi fingerprinting," in *2020 International Conference on Localization and GNSS (ICL-GNSS)*, 2020, pp. 1–6.
[19] J. Torres-Sospedra, P. Richter, A. Moreira, *et al.*, "A comprehensive and reproducible comparison of clustering and optimization rules in wi-fi fingerprinting," *IEEE Transactions on Mobile Computing*, 2020.
[20] R. Montoliu, E. Sansano, J. Torres-Sospedra, *et al.*, "Indoorloc platform: A public repository for comparing and evaluating indoor positioning systems," in *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2017, pp. 1–8.
[21] I. E. Radoi, D. Cirimpei, and V. Radu, "Localization systems repository: A platform for open-source localization systems and datasets," in *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2019, pp. 1–8.
[22] K. Kaji, K. Isomura, and T. Takai, "Step recognition method using air pressure sensor," in *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2019, pp. 1–8.
[23] C. Villien, A. Frassati, and B. Flament, "Evaluation of an indoor localization engine," in *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2019, pp. 1–8.
[24] "ISO/IEC 18305:2016 Information technology — Real time locating systems — Test and evaluation of localization and tracking systems," Standard, 2016.
[25] F. Potortì, A. Crivello, P. Barsocchi, *et al.*, "Evaluation of indoor localisation systems: Comments on the iso/iec 18305 standard," in *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, IEEE, 2018, pp. 1–7.
[26] J. Torres-Sospedra, R. Montoliu, S. Trilles, *et al.*, "Comprehensive analysis of distance and similarity measures for Wi-Fi fingerprinting indoor positioning systems," *Expert Systems with Applications*, vol. 42, no. 23, pp. 9263–9278, 2015.
[27] L. Klus, D. Quezada-Gaibor, J. Torres-Sospedra, *et al.*, "Rss fingerprinting dataset size reduction using feature-wise adaptive k-means clustering," in *2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, IEEE, 2020, pp. 195–200.
[28] P. Barsocchi, S. Chessa, F. Furfari, *et al.*, "Evaluating ambient assisted living solutions: The localization competition," *IEEE Pervasive Computing*, vol. 12, no. 4, pp. 72–79, 2013.