The Wisdom of 1,170 Teams: Lessons and Experiences from a Large Indoor Localization Competition

Yuming Hu[1], Xiubin Fan[2], Zhimeng Yin[2 7], Feng Qian[1], Zhe Ji[3], Yuanchao Shu[4], Yeqiang Han[3], Qiang Xu[3], Jie Liu[5], Paramvir Bahl[6]

[1]University of Minnesota [2]City University of Hong Kong [3]XYZ10 Technology [4]Zhejiang University
[5]Harbin Institute of Technology (Shenzhen) [6]Microsoft [7]CityU HK Shenzhen Research Institute

ABSTRACT

We organized an online fingerprint-based indoor localization competition in 2021. It attracted 1,170 teams worldwide. The teams were provided with a 60 GB dataset including WiFi, BLE, IMU, and geomagnetic field strength data collected from 204 buildings to build their localization algorithms, which were then evaluated against a separate test dataset. The competition received 28,009 submissions. The top team achieved an average accuracy of 1.50m. This paper reports the lessons we learned from analyzing the submissions, as well as our experiences in organizing the competition, through both qualitatively studying the teams' algorithms and quantitatively characterizing the competition results.

CCS CONCEPTS

• **Information systems** → **Location based services**; *Sensor networks*; Global positioning systems; • **Networks** → **Location based services**.

KEYWORDS

Indoor Localization; Fingerprinting; Online Competition.

ACM Reference Format:
Yuming Hu, Xiubin Fan, Zhimeng Yin, Feng Qian, Zhe Ji, Yuanchao Shu, Yeqiang Han, Qiang Xu, Jie Liu, Paramvir Bahl. 2023. The Wisdom of 1,170 Teams: Lessons and Experiences from a Large Indoor Localization Competition. In *The 29th Annual International Conference on Mobile Computing and Networking (ACM MobiCom '23), October 2–6, 2023, Madrid, Spain.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3570361.3592507

ACM Mobicom '23, October 2–6, 2023, Madrid, Spain

ACM ISBN 978-1-4503-9990-6/23/10…$15.00
https://doi.org/10.1145/3570361.3592507

1 INTRODUCTION

Indoor localization has been an active research topic for more than two decades. Among numerous technologies, *fingerprint-based* indoor localization has attracted the most attention in literature and was the first (and probably the only one) that has registered large-scale commercial deployment in recent years [3, 14, 23, 32]. A fingerprint-based localization system adopts a learning paradigm at its core. In the offline training phase, system operators survey the site to collect various features (*e.g.,* RSSI of radio signals) as well as the location ground truth, and use them to build a model. In the online localization phase, end users' devices (*e.g.,* smartphones) collect the same features and use the model to infer the current location. In practice, a well-designed system may involve much more sophisticated, inter-disciplinary mechanisms to improve the accuracy, usability, and resource efficiency, such as signal processing, sensor fusion, machine learning, crowdsourcing and edge/cloud offloading, to name a few.

In 2021, we organized an online indoor localization competition. This is to our knowledge the largest open-to-public indoor localization competition in terms of the data size, as well as the number of participants and received submissions. It attracted 1,446 contestants from 64 countries making up 1,170 teams. The participating teams were provided with a 60 GB dataset (with location ground truth) and their corresponding floor plans to train/build their solutions. The dataset was collected by professional surveyors hired by us from 204 diverse buildings. The fingerprints include WiFi RSSI, Bluetooth Low Energy (BLE) RSSI, inertial sensor data (IMU), and geomagnetic field strength (GMF). Over 4 months, the competition received 28,009 submissions, and the best team achieved an average localization accuracy of 1.50m.

Compared to individual localization projects and previous indoor localization competitions [25, 34], this online competition has its unique advantages: it reaches out to a much wider range of audiences with diverse background and expertise, and provide a platform to evaluate various solutions using the identical benchmarks from real-world building environments. By comparing the "crowd-sourced" solutions side by side, we can get an unbiased, comprehensive view that helps

us understand the recent advances in fingerprint-based indoor localization, and offers critical insights for improving the state-of-the-art. From the participants' perspective, an online competition also provides an ideal venue where they can exchange ideas and share experiences. Overall, we feel that our efforts were well paid off in particular given that the research community and public benchmarks of indoor localization are far less mature than many other domains (*e.g.,* image recognition in machine learning).

This paper reports the lessons we learned from analyzing the submissions, as well as experiences in organizing the competition. Specifically, we would like to answer the following questions. (1) What constitutes the overall architecture of a successful localization solution? (2) How much can deep learning, which is recently being fused into numerous mobile computing applications, help improve the localization accuracy? (3) What is the accuracy limit of fingerprint-based localization, which, we believe, can be approximated by the very top ones among more than 20,000 submissions? (4) What are the common errors (*i.e.,* poor localization accuracy) made by most teams? (5) How to improve the design of future competitions?

We face two major challenges when attempting to answer the above questions. The first challenge stems from the *high complexity* of the teams' solutions (*i.e.,* their localization logic). Modern fingerprint-based localization systems are highly sophisticated. Multiple modules including positioning (absolute localization based on the current fingerprint), tracking (relative localization based on the previous trajectory), pre/post processing (*e.g.,* using floor plan to make corrections), floor estimation, online (re)training, to name a few, need to work together to deliver accurate results [2, 12, 20, 48, 57, 62]. Many modules also own large parameter spaces. While we observe all these modules from many teams' solutions, fully understanding how they work individually and collectively poses a major difficulty. To make it worse, we usually have no access to the teams' source code.[1] To address this challenge, we *quantitatively* characterize the submitted localization results, whereas *qualitatively* study the teams' localization logic, by leveraging multiple venues: we encourage teams to disclose their algorithms or even source code on the competition website after the event; we examine the commit history and its comment section to learn the improvements made by teams over four months; we also check posts in the competition forum and teams' blogs to obtain their high-level design decisions. Finally, after the competition, we conducted an online panel discussion and invited the top teams to share their solutions and experiences. The second challenge comes from the large number of the teams. To make our analysis tractable, we focus on analyzing the top 18 teams' solutions, which were found to be already highly heterogeneous and thought-provoking. Meanwhile, to avoid losing the big picture, we also provide general statistics of all teams' solutions.

Next, we summarize key findings and lessons from this large-scale competition.

• Among the top 18 teams, 15 were from industry as opposed to academia. Almost all 18 teams' solutions include positioning, tracking, and optimization modules that vary widely. The core positioning algorithm ranges from simple machine learning (ML) such as KNN to sophisticated deep learning such as CNN and LSTM. The tracking algorithms are also heterogeneous, from traditional dead reckoning to deep learning.

• Regarding the localization accuracy, the top 18 teams achieve an average accuracy of no more than 3.7m (a median of no more than 2.6m), sufficient for typical use cases of smartphone-based indoor localization. The team who won the first place achieved an average localization accuracy of 1.50m, which is close to even outperforming the state-of-the-art accuracy reported by academic publications [5, 21, 31] (using similar fingerprints, albeit in much smaller-scale lab settings). We believe this approximates the limit of fingerprint-based indoor localization using WiFi/BLE/IMU/GMF signals. Meanwhile, fingerprint-based approaches work reasonably well with simple engineering. Even for the 500th team, its mean positioning error can reach 5.7m – still acceptable for many indoor localization applications such as store/room navigation in malls/office buildings.

• For positioning (*i.e.,* absolute localization), parameter tuning is more important than model selection. With fine tuning for a given floor or building, simple machine learning algorithms can outperform deep learning algorithms. In fact, somewhat surprisingly, we find that the top three teams all adopt KNN as a main component in their positioning models.

• To achieve a high rank, in addition to good positioning algorithms, accurate tracking (*i.e.,* relative localization) is indispensable. We find that a solution's final score is more correlated with its tracking accuracy (Pearson correlation coefficient: 0.78) than its positioning accuracy (Pearson coefficient: 0.34). Deep neural networks can effectively reduce the tracking error by about 50% compared to the default dead reckoning routines we provided.

• The top-3 teams' success mostly owes to three designs: (1) lightweight positioning models judiciously customized to each site/floor; (2) learning-based tracking with high accuracy; (3) using floor plans to correct predicted trajectories.

[1] This is because we only require the teams to submit answers (*i.e.,* location coordinates) for given trajectories in the test dataset. Running the teams' code by ourselves will make grading very complex due to heterogeneous development environments and programming languages used by the teams.

• Large localization errors typically occur in corners and dead ends in buildings. Floor estimations are usually quite accurate, and their errors usually occur in the atrium area (a large, open multi-storied space in a large building).

• Top teams attempt to leverage various techniques to reduce their errors. While some techniques are expected (*e.g.,* leveraging the floor plan information for error correction), some approaches are unexpected. For example, a few top teams managed to "reverse engineer" the dataset to infer the device types we used to collect the data (*e.g.,* through the IMU sampling interval), and use them as a feature in the ML model.

• In addition to the above, we also learned valuable lessons on organizing localization competitions. For example, future competitions should minimize information leaks from side channels, make test sets more diverse, avoid using key landmarks as test points (to prevent easy guesses of answers), and employ more cross-floor traces to emulate more real-world usage scenarios. We report these experiences in §5.

Ethical Consideration. No data used in the competition or in this paper contains personally identifiable information (PII). All the analyses performed in this paper were based on public information or information voluntarily provided by the teams, as acknowledged by the teams and us as part of the competition rules. The top teams that won this competition were offered monetary awards. No team's solution was used in commercial products, nor do we intend to in the future.

Data Release. We have released the entire dataset (60 GB), containing WiFi, BLE, GMF, IMU readings and groundtruth locations collected by surveyors. The dataset URL is:

https://aka.ms/location20dataset

2 DESCRIPTION OF THE COMPETITION

This section describes the competition rules, its datasets, grading procedure, and the participating teams. We also compare our competition with a prior indoor localization competition held in 2014 [34].

2.1 An Overview of the Competition

In 2021, we organized an online indoor localization competition that lasted about 4 months. The participating teams were provided with a large scale real-world indoor dataset collected by smartphones, which comprise WiFi/BLE radio signals, inertial sensor data, and geomagnetic fields (GMF). They also had the access to the site information, including the floor plan and point of interest (POI). Besides, we provided them with the default pedestrian dead reckoning (PDR) routines, which can count the steps of the target and calculate the step length/heading.

The goal of each team is to predict the location (*i.e.,* floor and coordinate) of the target in the building. Note that each building has multiple floors. We collected large-scale traces as the training set, containing the indoor fingerprints and the ground truth locations of targets. For some traces, we removed the ground truth locations, and used them as the test set. The test trace consists of a sequence of fingerprints corresponding to a moving trajectory as well as a sequence of timestamps $\{t_i\}$. The detailed characteristics of the selected trajectories will be shown in §2.2 and §2.3. For each test trace, a team is asked to submit its answer, which consists of a sequence of locations $\{l_i\}$, where l_i is the estimated location at t_i.

The competition received 28,009 submissions. Figure 1 plots the best score across all submissions and the number of submissions per day over the course of the competition. The teams only needed to submit answers (*i.e.,* localization results) to given traces as input queries. As shown in Figure 1, the submissions gradually increased during the competition, and surged in the last 7 days. The best score gradually decreased since February and reduced significantly in the last week. Note that the score is a combination of positioning error and floor error (the lower the better, see §2.3). Finally, the best score has reached a localization accuracy of 1.50m. The top 3 teams received 5,000, 3,000, and 2,000 U.S. dollars, respectively, as monetary rewards.

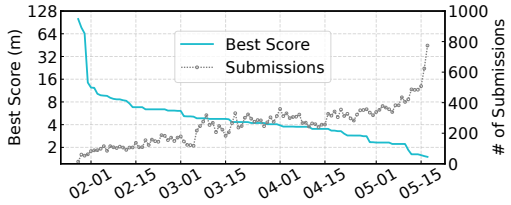


Figure 1: Competition results.

2.2 The Datasets

The participants in this competition were provided with a 60GB fingerprint dataset of 204 buildings, most of which are shopping malls in China. Since we aimed to provide indoor localization for smartphone users, the fingerprint collection was also based on smartphones. We developed a custom mobile application for data scanning, and hired professional surveyors to collect the indoor fingerprints of the buildings. The data were collected from 2018 to 2020, and contained about 30,000 traces with a total distance of 2,257 km.

A challenge here is that we do not know the exact location of the surveyor (*i.e.,* the ground truth). We use visually recognizable landmarks (*e.g.,* pillar, door) in the building as anchors. As illustrated in Figure 2, we generate waypoints in walkable areas based on the floor plan. Surveyors were asked to leverage these waypoints to plan their walking trajectories, during which they will scan data through smartphones and store them in the trace file. Surveyors can also modify

the waypoints as needed to make data collection easier. For a trace containing several waypoints, the surveyor walks from the first point to the last point, and his/her smartphone will record the fingerprints. When the surveyor passes a waypoint, he/she will mark the current waypoint in the collection application as the location ground truth.

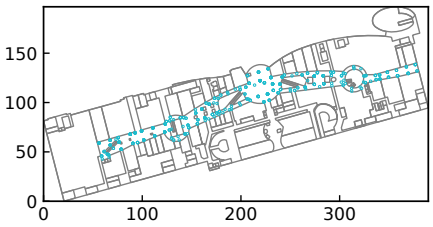


Figure 2: A sample of waypoints on the floor.

The fingerprint includes inertial measurement unit (IMU) reading, radio signals, and geomagnetic field strength (GMF). The IMU data consists of accelerometer and gyroscope. The radio signals include both WiFi RSSI and Bluetooth Low Energy (BLE) RSSI (if available). In addition, we provide the metadata of each floor (image, size, floor plan, and POI data).

The test traces are selected based on the real-world user trace length, and contain 626 traces at 26 sites. We removed the location information of the waypoints (*i.e.,* test points) in each test trace, and the teams needed to estimate the floor and coordinate of each test point based on the provided fingerprints.

2.3 Grading Submitted Solutions

To avoid overfitting, the evaluation of each team's submission was divided into two parts, including a public and a private part [4]. During the competition, the teams only knew the public leaderboard, in which the score of each team was calculated based on 15% of the test data. The private leaderboard, which was determined by the other 85% of the test data, remained confidential until the end of the competition. The final ranking was based on the private leaderboard, which was oftentimes different from the public leaderboard. We only revealed the private leaderboard (and thus the final ranking) after the competition.

In this competition, we primarily focus on the accuracy metric, and leave assessing other metrics such as power consumption and localization delay as future work (§5). We consider both positioning error and floor estimation error to calculate the final score as follows:

$$score = \frac{1}{N}\sum_{i=1}^{N}\left(\sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2} + p \cdot |\hat{f}_i - f_i|\right) \quad (1)$$

where N is the number of waypoints in the test set, $(\hat{x}_i, \hat{y}_i)$ are the predicted locations for a given waypoint, (x_i, y_i) are the ground truth location for a given waypoint, p is the floor penalty (set to 15), and $\hat{f}_i, f_i$ are the predicted and ground-truth floor level (an integer) of a waypoint, respectively.

The length of the traces in the test set is illustrated in Figure 3. We select the test trace length based on a separate study of real users' mobility in real buildings (references omitted for anonymization). It is usually not easy to estimate the initial position of the target due to limited fingerprints. Furthermore, the accumulation of positioning errors might lead to large errors in the long traces. Therefore, most of our test traces (>80%) are in the range of 40–150 meters. We also take a small selection of short (<40 meters) and long traces (>150 meters) to test the robustness of each team's solution.

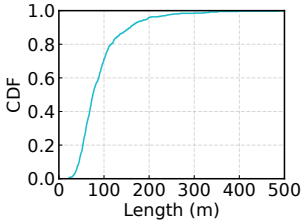


Figure 3: The length of the traces in the test set.

This competition is not real-time. Each team only needs to make offline estimations and submit their answers to our system (up to 5 times per day). The whole dataset, including the training set (with ground truth) and the test set (without ground truth), was provided to the teams in advance. This design makes the grading system very easy to implement and the competition rules easy to follow. It also helps scale up the competition. Nevertheless, since the teams had the fingerprints of the entire traces, they could possibly leverage future information to correct their estimates of the previous test points, especially for long traces (some real-world localization systems perform similar optimizations of using the current location to correct past trajectories [49]). We leave holding large-scale real-time localization competitions as future work.

2.4 Participating Teams

The competition attracted a total of 1,170 teams from all over the world. 80% of the top 20 teams, and 60% of the top 50 teams, were from industry. Many participants chose to team up (up to 5 members per team) so they could develop sophisticated solutions. For example, most of the top 20 teams leveraged ensemble learning that makes use of different algorithms developed by their members. Each team could submit up to 5 solutions per day during the 4-month competition, and the top 50 teams made an average of more than 200 submissions.

The basic statistics of the top 18 teams are listed in Table 1. (Some data are missing and we mark them with "-".) The teams proposed various approaches to achieve accurate

Table 1: Top 18 teams that participate in the competition. The teams are listed in order of the private scores (85% of the test set). The score is the weighted average of localization error and floor error.

Rank	Background	Model				Score	
		Floor	Positioning	Tracking	Generalization	Public	Private
1	Industry	-	KNN & LGBM	CNN	F	0.89	1.50
2	Industry	-	KNN, MLP & RNN	LSTM, PDR & MLP	F	1.39	2.20
3	Industry	-	KNN	GLM	F	2.04	2.49
4	Industry	-	-	-	-	2.25	2.68
5	Industry	CNN	CNN	CNN	F	2.40	2.77
6	Industry	-	-	-	-	2.01	2.82
7	Industry	MLP	LSTM & LGBM	WaveNet	-	2.47	2.83
8	Industry	-	LSTM	CNN	T	2.53	3.07
9	Industry	No	GRU, LSTM & CNN	GRU & LSTM	T	2.54	3.11
10	Industry	No	RNN	RNN	-	2.57	3.12
11	Industry	No	LSTM	PDR	F	2.01	3.16
12	Industry	KNN	KNN	PDR	F	2.69	3.20
13	Academia	LSTM	LSTM & MLP	MLP	T	2.55	3.36
14	Industry	-	LGBM	-	-	2.65	3.54
15	Academia	KNN	LSTM & CNN	LGBM & PDR	T	2.91	3.54
16	Industry	LGBM	LSTM & MLP	-	T	3.00	3.56
17	Academia	-	LSTM	-	-	3.19	3.69
18	Industry	-	LGBM & MLP	MLP & CNN	F	2.75	3.74

indoor localization, including KNN [1], LGBM [27], MLP [40], CNN [30], WaveNet [51], RNN [41], LSTM [22], BiLSTM [17], GRU [7], *etc.* They usually calculated the absolute and relative locations of the target separately, and then combined them to predict the final location. For both positioning (absolute localization) and tracking (relative localization), most teams preferred to use neural network (NN)-based approaches. Besides, they tended to use more than one model to predict the target's location, and leverage ensemble learning to improve the localization accuracy. The best scores reached 0.89m and 1.50m (lower is better) on public and private test sets, respectively.

2.5 Comparison to the 2014 Competition

In 2014, Microsoft held an indoor localization competition [34] at a hotel in Berlin. The competition attracted 22 teams from the world, most of which (17 of 22) came from academia. They mostly employed traditional (non-NN) methods to estimate the location of their targets, and it took them months to years to build their systems. There were two types of solutions, infrastructure-free and infrastructure-based. The infrastructure-free approach used existing features (*e.g.,* WiFi, GMF) to achieve indoor localization, whereas infrastructure-based approach required deploying additional hardware. Finally, infrastructure-based solutions achieved an average positioning error of 0.72m, and infrastructure-free solutions could reach 1.56m.

Our competition is infrastructure-free only. It bears a much larger scale in terms of the number of buildings and participating teams. Regarding the solutions, we observe a major paradigm shift: most teams chose to leverage neural network (NN)-based methods, and it took them less than 4 months to build and tune their models. Despite that, somewhat surprisingly, we observe no noticeable improvement in the accuracy for infrastructure-free approaches. The best teams have achieved an average accuracy of 1.50m on the private test set (taking into account the floor penalty) and 0.89m on the public test set.

■ **Finding 1.** *Compared to the Microsoft competition [34] in 2014, most teams in our competition adopt neural network (NN)-based models. Despite the evolution of learning algorithms, after seven years, however, there appears no significant improvement of positioning accuracy. Fingerprint-based approaches have likely reached the limit of their potentials. Nevertheless, the top 18 teams can all achieve an average accuracy of <3.7m, sufficient for most everyday use. Surprisingly, the top 3 teams employ non-NN algorithms as the main component of their positioning models, achieving an accuracy of <2.5m.*

3 ANALYSIS OF TEAMS' SOLUTIONS

In this section, we take a close examination of the teams' solutions, focusing on the top 18 teams.

3.1 Assessment Methodology

Recall from §1 that we face a major challenge that we do not have the code of each team. This is because the teams were only requested to submit their prediction results during the competition. Although some competitors have voluntarily released their high-level designs, their implementation details remain unknown, let alone the details of their algorithm evolution. To overcome this challenge, we investigated multiple sources, including blogs, commit history, forum posts, and source code (if shared publicly). To ensure our obtained teams' algorithm logic is correct, we perform thorough cross-checks using the above sources. All the results presented in the remainder of paper were cross-checked from at least two sources. In addition, except for the forum posts, all the other information was either not available to other teams during the competition or published after the competition, so there is little incentive for teams to falsify their posted information. We therefore believe our results are convincing and derived through a scientific approach. Specifically, (1) Many of the top 18 teams disclosed their logic after the competition. Based on this, we manage to obtain a reasonably good overall picture of their design and key tradeoffs they balance. (2) Some teams specified the module and algorithm in their submission history. Recall from §2.4 that each team was allowed to submit up to 5 times per day. We have recorded all the submitted files and their scores, and kept the prediction results for each test point. Many submissions had names with various "meta data" (*e.g.,* algorithm name) that indicate their corresponding solution versions. We can thus calculate the performance improvement brought by each module by comparing the scores of different versions of the submission file. (3) Competitors were encouraged to discuss the issues they encountered and the corresponding solutions during the competition. Therefore, we can know the algorithms they adopted and their roles in the overall solution. Besides, a small group of competitors noted in their discussions the dates they incorporated certain techniques into their models. Correlating this information with the submission history offers us more detailed information. (4) We invited top teams to a workshop and panel discussions to learn their solution. (5) Some teams have also released part of their source code.

Using the above approaches, we can *qualitatively* learn the solution structure, input, output, algorithm of each module, and the corresponding performance improvement brought by each module. Furthermore, we are able to *quantitatively* evaluate the performance of each submission and assess the impact of the delta across submissions. Note that many teams did not publish their models or discuss them with others. Fortunately, almost all of the top 18 teams, which we focus next, have shared their solutions through one or more channels described above.

3.2 An Overview of Solutions

Figure 4 illustrates the basic architecture of most (top-18) solutions, including positioning, tracking, and various optimization modules. There may not be a separate floor estimation model, as some positioning models can also predict floor levels. Positioning aims to estimate the absolute location of a target, usually using radio signals (§3.3). The scanning frequency of radio signals on smartphones is oftentimes limited (*e.g.,* 4 times per 2 minutes [15]), and there might not be radio signals in some areas, thus we cannot always achieve absolute localization. Therefore, in addition to positioning, we need tracking, which uses inertial data to calculate the walking distance and direction of the target. In our competition, participants usually leveraged tracking algorithms to estimate the distance between two test points (waypoints, see §3.4). In addition, top teams often employed various post-processing techniques. For example, most competitors strategically combined positioning and tracking results to improve the accuracy.

Table 2: Correlation between modules and final score.

Factors	Input type	Positioning error	Tracking error	Optimization method
Correlation	0.31	0.34	0.78	0.40

Table 3: Improvement brought by each module.

Modules	Ensembling	Combine Pos.&Tck.	Snap-to-landmark	Device ID
Imp. (m)	0.02–0.2	0.4–0.9	0.3–1.4	0.03–0.4

Table 2 shows the Pearson correlation coefficients between {input type, positioning error, tracking error, optimization method} and final score across the top 18 teams, which are {0.31, 0.34, 0.78 and 0.4}, respectively. The input type (*e.g.,* WiFi, BLE) and optimization method (*e.g.,* snap-to-landmark) are non-numeric, therefore we binary encode their entries. We find that tracking error is highly correlated with the final score. The importance of tracking will be further detailed in §3.4.

Table 3 illustrates the improvement that each module brings to the positioning models of the top 18 teams. Ensembling, combining positioning and tracking, snap-to-landmark, and device ID can enhance the localization accuracy by 0.1m, 0.6m, 0.8m and 0.3m on average, respectively. We will detail these components in §3.3, §3.4, §3.5.

3.3 Positioning (Absolute Localization)

Positioning models are designed to locate the target using the currently observed fingerprint. The fingerprint signals

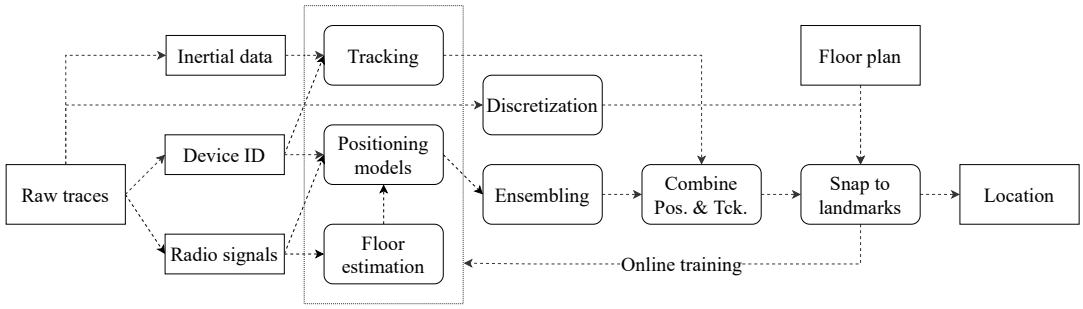


Figure 4: The basic procedure of most top teams' solutions. Most teams would build positioning and tracking models separately, combine them by using linear methods, and then leverage optimization techniques to improve the localization accuracy. Device ID is inferred by reverse engineering, not explicitly given.

include WiFi, BLE, and GMF. Since only a subset of buildings in our dataset had BLE beacons deployed, most teams only used WiFi signals for positioning. In the top 18 teams, only the team ranked 1st, 8th, 13th and 14th incorporated BLE signals into their models (the information of the 4th and 6th teams is missing). In addition, only the 1st team used geomagnetic fields (GMF) for positioning, and they claimed GMF did little to boost the localization accuracy. Most teams used cosine similarity and Pearson correlation (as opposed to Euclidean distance) for feature matching.

Solution Categories. We categorize the positioning solutions into two types, namely the non-neural network (non-NN) and NN-based ones.

• Non-NN solutions include k-nearest neighbors (KNN) and decision trees (top teams use LGBM).

• NN-based solutions comprise recurrent neural network (RNN) and its variants (*e.g.,* LSTM), multilayer perceptron (MLP), and convolutional neural network (CNN) and its variance (*e.g.,* WaveNet).

Most of the top 18 teams adopted NN-based models. The most popular positioning model is LSTM, which can deal with sequence data. Although CNN is typically used to handle image data, it reaches up to the 5th place. Surprisingly, the method that performs best turns out to be KNN, a lightweight non-NN solution. KNN is considered a lazy learner [67] since there is no training in KNN. It is also easier to implement, update, and interpret compared to NN models. Furthermore, lightweight models such as KNN are more resource- and energy-efficient than NN-based models. The top-3 teams mainly employed KNN as their positioning model, and the 3rd-place team only used KNN. Another non-NN model used by top teams is LGBM, which is built upon decision trees. The top-1 team combined KNN and LGBM to build their positioning model. Other popular non-NN algorithms in the literature such as particle filtering [10] were not used by top teams.

Model Generalization. In indoor environments, radio signals are often influenced by a building's layout, materials, *etc.* Therefore, it is difficult to derive a general model for signal fading and propagation. Most of the top teams (*e.g.,* top-5 teams) therefore adopted per-site models in this competition, as shown in the "Generalization" column in Table 1. Nevertheless, 5 out of the 18 top teams chose to train a single model for all sites, using NN-based models. These general models also performed reasonably well despite not ranking at the top.

An important prerequisite for lightweight non-NN models is to tune the parameters for each site and even each floor, as it is difficult to generalize such models to all sites: the number of neighbors and the weight of each neighbor in weighted KNN models may vary across sites and floors. This incurs additional overhead for parameter tuning, but turns out to be worthwhile, as indicated by the competition results.

■ **Finding 2.** *Lightweight models (*e.g., *KNN) can achieve high positioning accuracy and outperform neural network-based solutions. The top three teams use KNN as the main component of their positioning models, with some hyper-parameters carefully tuned for each site/floor.*

Ensemble Learning. Ensemble learning leverages many weak learners to achieve better prediction results. We expected to see significant improvements in positioning accuracy when combining multiple learning algorithms. However, the results suggest the opposite. For top teams, they have already trained strong learners using multiple input features and/or deep neural networks. After building these individual models, most top teams directly used the weighted average of the models' output to estimate the target's location. Ensembling these strong learners brings little improvement (< 0.1 m) in average localization accuracy compared to the team's best individual model.

■ **Finding 3.** *Ensemble learning does not make a significant performance boost (<0.1m) to the top-ranked teams' solutions. This is because they have already trained strong learners.*

3.4 Tracking (Relative Localization)

Due to hardware limitations and energy-saving considerations, most smartphones infrequently scan the radio signal (*e.g.,* twice in one minute for WiFi scan [15]). This is reflected in our dataset, as there might be no WiFi/BLE scanning activity (and therefore no radio signal) as the surveyor walks past a test point. In contrast, the IMU samples have a much higher frequency (50 HZ in our dataset). This renders tracking, which uses IMU data for continuous, relative localization, highly important. We find that some teams used IMU data to interpolate radio fingerprints at the test points, whereas some directly applied Pedestrian Dead Reckoning (PDR) to infer a test point's location from a previous location estimated from radio signatures (*i.e.,* positioning). Recall from Table 2 that the importance of tracking is also reflected in its high correlation with the final score (0.78).

One question faced by the teams is whether to combine positioning and tracking into a single model, or keep them separate. We find that almost all the top teams chose the latter: they built positioning models and tracking models separately, and then combined their outputs to get the final localization result. A possible explanation is that, since the relationship between relative and absolute positioning can be analytically reasoned, there is no need to mix them into, for example, a neural network model.

Recall that the teams were provided with a default tracking model (*i.e.,* a simple PDR implementation). Nevertheless, most top teams chose to build their tracking models based on neural network (NN), or use NN to train the hyperparameters of the default model. We find that compared to the default PDR implementation we provided, NN-based methods can reduce the tracking error by up 60%.

■ **Finding 4.** *Many teams spent great effort on developing positioning models. However, they did not achieve high rankings due to a lack of accurate tracking models. As shown in Table 2, the Pearson correlation between tracking results and the final score is as high as 0.78. To achieve high tracking accuracy, top teams adopt neural network models that reduce the errors by up to 60% compared to traditional PDR methods.*

Combining Positioning and Tracking Results. Tracking is also used by many teams to estimate the next test point's location from the previous test point (whose location has already been inferred). Then a problem faced by these teams is how to combine the positioning and tracking results. As a classical problem in robotics, it can be solved using methods such as particle filtering [10] and Kalman filtering [8]. However, these methods are not easy to implement. Interestingly, we find that most top teams solve them using a simple linear model as follows. Assume that there are two consecutive waypoints X_i and X_{i+1}. Their locations estimated based on radio signals are denoted as $\hat{X}_i$ and $\hat{X}_{i+1}$, and the displacement from X_i to X_{i+1} calculated based on IMU data (*i.e.,* tracking) is denoted as $\Delta\hat{X}_i$. We then want to minimize the weighted sum of positioning error and tracking error over the N test points. That is,

$$minimize \sum_{i=1}^{N} \alpha_i \|X_i - \hat{X}_i\|^2 + \sum_{i=1}^{N-1} \beta_i \|(X_{i+1} - X_i) - \Delta\hat{X}_i\|^2 \quad (2)$$

Note that Equation 2 just exemplifies a simple cost function; many teams add other components to it. The optimization is easy to solve and has a closed-form solution. For the weights (α and β), some teams empirically pick their values for all the sites, whereas some teams further use machine learning to train them for different sites or even floors.

■ **Finding 5.** *There is no need to build complex methods to combine positioning and tracking models. Top teams choose to combine them using the simple linear method shown in Equation 2. It has a closed-form solution, and its parameters can be trained through machine learning, leading to satisfactory localization results.*

3.5 Other Optimization Techniques

Top teams employ many optimization techniques to improve their localization accuracy. We find that they can be grouped into four categories, namely *snapping to nearby landmarks*, *device identification*, *using floor plan*, and *online training*. Based on the available data (Table 3), snapping to landmark and device identification can boost the localization accuracy by 0.8 m and 0.3 m, respectively, on average. It is difficult for us to quantify the benefits of the other two optimizations due to insufficient data.

Snapping to Nearby Landmarks. Our surveyors oftentimes use visually recognizable landmarks (*e.g.,* pillars and doors) as waypoints because of their convenience. The same applies to the waypoints in the test dataset because the test set was sampled from the surveyed waypoints. We find that a few top teams "snap" their localization results to nearby landmarks, given that landmark locations are more likely to be a test point's answer compared to non-landmark locations. Some teams even manually label the landmarks using the provided floor plan to increase the chance of capturing landmarks in the test set. This "optimization" may not be effective in real-world usage scenarios because a real user may invoke localization services at any location. Future localization competitions should thus add more non-landmark locations to the test dataset.

Device Identification. We find that a few teams managed to "reverse engineer" the dataset to infer the devices we used to collect the data. Figure 5 shows one example of using the IMU sampling interval as the (pseudo) device ID. We set the sampling frequency of all surveyors' smartphones

to 50 HZ. However, due to their hardware differences and/or manufacturing variations, there exist subtle but steady differences among the devices, which can be distinguished by identifying the "cliffs" in the sampling interval distribution observed from the data. Similarly, the hard iron distortion of magnetometers [13] was used by some teams as a smartphone identifier. The (pseudo) device ID was then used by the teams as a feature of their positioning and tracking models to improve the accuracy. Note that this approach can also be used in real-world localization systems by directly capturing the (real) device ID/model information.

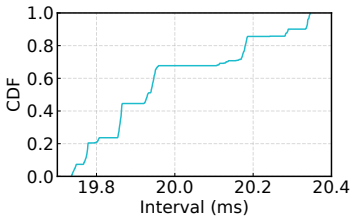


Figure 5: The interval between accelerometer readings.

■ **Finding 6.** *The organizer did not provide device IDs in the dataset. However, we find that some high-ranking teams "reverse engineered" the dataset to infer the pseudo ID of each device (*e.g., *through IMU sampling interval and hard iron distortion of magnetometers). Using the pseudo ID as a feature helped further improve the positioning accuracy of top teams by 0.3m. This confirms the big impact of device heterogeneity on localization accuracy [5] in the wild.*

Using Floor Plan. We find that many top teams leveraged the floor plan to correct the localization results. Due to the constraint of the building layout, the areas accessible by a target are limited. If the estimated moving trajectory of the target fully or partially falls outside the walkable area, the teams use various heuristics to shift the trajectory back to the walkable area based on the floor plan. This optimization is performed as a post-processing step after positioning and tracking. It is well applicable to real-world localization systems if the floor plan information is available.

Online Training. Several top teams (re)trained their models online. Specifically, if a localization result is considered reliable, the result and its corresponding fingerprints will be added to the training set, and the model will be retrained. We find that the teams used various heuristics to determine the reliability. For example, the localization result of a waypoint is considered reliable if its timestamp is close to the time when the radio scan activity occurs. In another example, when ensemble learning is used, a result is regarded as reliable if the variance of all learners' outputs is small. We observe that the teams have applied online training to multiple modules such as positioning, tracking, and floor estimation.

3.6 Floor Estimation

There are two approaches for floor estimation. (1) Floor estimation is integrated into the positioning module. (2) There is a standalone model to predict the floor level. We find that both approaches worked very well. As shown in Table 1, among the 3 teams taking Approach (1), only 1 team (10th place) has a floor prediction error of 0.38%, while the other 2 teams have no floor error. Among the 6 teams taking Approach (2), only 1 team (13th place) has a floor error of 0.18%, while the other 5 teams have no floor error. Floor prediction models are usually more lightweight than the positioning models. More discussions on floor estimation accuracy and the limitation imposed by our competition can be found in §4.4.

3.7 A Case Study of the Top Three Teams

We conduct a case study of the top 3 teams to investigate why they ranked very top. We find that their key efforts mostly consist of the following. (1) Their positioning models were carefully tuned for each site and even floor. Although the models were based on lightweight algorithms (*e.g.,* KNN), their performance is comparable to or even better than neural networks after fine-tuning. (2) Among the top 18 teams, the top 3 teams achieved the highest tracking accuracy. Their tracking models were mainly learning-based, helping reduce up to 60% of errors compared to the default PDR design we provided. (3) They all leveraged floor plans to correct erroneous trajectories by shifting them from unwalkable areas to corridors, as described in §3.5. Note that the top-3 teams also employed other optimization techniques (*e.g.,* snapping to nearby landmarks, §3.5) to boost the localization accuracy. However, these optimizations with similar logic were also used by many of the other top teams (ranked from 4th to 18th). Therefore, they are unlikely the main contributors of the top-3 teams' success.

■ **Finding 7.** *The top-3 teams' success mostly owes to three designs: (1) lightweight positioning models judiciously customized to each site/floor; (2) learning-based tracking with high accuracy; (3) leveraging floor plans to correct predicted trajectories. All three strategies are applicable to real-world indoor localization systems.*

4 CHARACTERIZING SOLUTIONS' LOCALIZATION ACCURACY

This section complements our qualitative analysis in §3 by quantitatively characterizing the accuracy of the teams' solutions. In addition, we extend our analysis to the top 50 teams unless otherwise noted.

4.1 Error Distribution across Teams

We first describe how localization error is calculated. For a trace with n waypoints (ground truth) $\{A_0, ..., A_{n-1}\}$, assuming that their localization results are $\{B_0, ..., B_{n-1}\}$, then $|A_iB_i|$ is referred to as the *final* localization error of A_i. This is the error calculated based on the teams' submitted answers. In addition, we also define the *relative* localization error to estimate the teams' tracking inaccuracy. To calculate the relative localization error, we translate B_i to $B_i + \overrightarrow{B_0A_0}$ (denoted as B_i'), *i.e.,* B_0 is shifted to the ground truth A_0. Then $|A_iB_i'|$ represents the relative localization error of A_i. If unspecified, the localization error refers to the final localization error in the remainder of this subsection.

Overall Score Distribution. The score of each team is calculated based on Equation 1, and the results are demonstrated in Figure 6a, which is generated based on the private test set (§2.3). Overall, among the top 50 teams, the performance gap between the top-ranked teams is greater than that of the bottom-ranked teams. Specifically, we find that there are three score clusters among the top 50 teams. The top 5 teams form the first cluster of the leader board, and their scores differ significantly from the other 45 teams. They have achieved a mean score of 2.33m with a standard deviation of 0.46m. The mean of their first-order difference (*i.e.,* difference between two scores ranked next to each other) is 0.32m, and the difference between the top 2 teams is as high as 0.7m. The second score cluster consists of the teams ranked 6–15, who have a mean score of 3.17m with a standard error of 0.24m. Their mean first-order difference is 0.08m. The third score cluster comprises teams ranked 16–50, who have an average score of 4.24m with a standard deviation of 0.32m. Their average first-order difference of 0.03m is the smallest of the three clusters.

Final vs. Relative Error Distribution. Figure 6b plots the final localization errors across the top 50 teams. We do not take into account the floor estimation error here, and the localization errors represent the mean prediction error of all traces, including both public and private test sets. The mean accuracy of all 50 teams is 3.67m, and the best accuracy can reach 1.33m. The overall distribution is similar to that in Figure 6a. Figure 6c plots the relative localization errors across teams. Recall that a relative location corresponds to the location relative to the anchor point, which in our case is the initial location ground truth. The mean relative localization error is 4.73m, and the minimum relative error is 2.10m, which is achieved by the 1st ranked team. Compared with the final localization errors in Figure 6b, the relative localization errors are approximately 1m larger. In addition, the relative error is much more fluctuating than the final error. The above results suggest that achieving accurate tracking is challenging. Meanwhile, as shown in Table 2, tracking remains critical for determining a team's rank.

Table 4: Locations of test traces with mean errors > 9m.

Error Location	Dead End	Corner	Other
Ratio	33.3%	27.8%	38.9%

4.2 Error Distribution across Traces

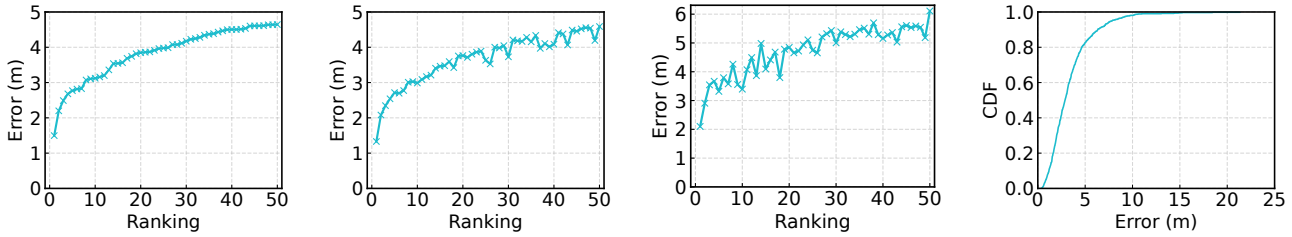
The localization error depends on many factors. In this subsection, we explore the localization accuracy across different traces (trajectories) that represent diverse environments in a building.

Figure 6d plots the distribution of (final) localization errors over all 626 test traces used to grading. For each trace, we calculate its localization error of all the top 50 teams and take their average as the localization error. The median and mean error for these test traces are 2.87m and 3.41m, respectively. Figure 6d suggests a long-tail distribution: most traces have an accurate localization result, but there is a long tail of errors. For example, the 25th percentile of the estimation error is only 1.83m, while the 75th percentile can reach 4.24m, not to mention that the maximum error can reach up to 21.38m.

To dig into the root causes, we categorize the locations where large errors occur, and show the results in Table 4. Here we define localization errors being greater than 9.0m as large errors, which account for 2.9% of the test traces. Note that we do not take the floor estimation error into consideration here. We then classify the large-error traces into three categories, namely dead end[2], corner, and other (also referred to as default areas). Approximately 33.3% of the traces with large localization errors are located at dead ends, where there are fewer strong radio signals compared to default areas. Some traces are even in unused spaces where there are no WiFi APs nearby. 27.8% of the large errors occur in a building's corners. This is because signal strength attenuation in corners is more dynamic and complex than that in open areas due to excessive signal reflection. The remaining 37.8% of the large errors happen in default areas due to various reasons. In contrast, within all the test traces (regardless of their mean errors), more than 80% belong to default areas. Overall, large errors in dead ends and corners are caused by a lack of strong signals and the complex attenuation (multipath fading) incurred by indoor layouts.

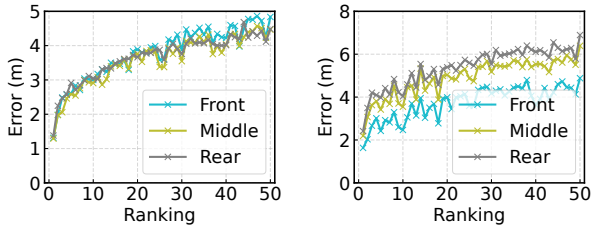
■ **Finding 8.** *More than 60% of test traces with large localization errors (mean error > 9m) belong to dead ends and corner areas due to a lack of strong radio signals and more complex signal attenuation compared to default areas.*

[2]If more than 50% of the test points within a trace belong to dead ends or corners, we classify the trace as the dead end or corner category, respectively. The remaining traces are classified as "Other".



(a) Score (b) Final Error (c) Relative Error (d) Error Distribution

Figure 6: The localization results of the top 50 teams. The score is calculated based on Equation 1.



(a) Final Error (b) Relative Error

Figure 7: localization errors of different trace segments.

4.3 Error Distribution across Segments

We now investigate how errors occur at different stages within a trace (trajectory). We divide each test trace into three *segments*: front, middle, and rear, which consist of the first 1/3, the second 1/3, and the last 1/3 of the trace's test points, respectively. For each segment, we define its localization error as the average error across all its test points. Figure 7 plots the localization errors of the top-50 teams across all front, middle, and rear segments, for both final and relative localization. In terms of final localization errors (Figure 7a), there is no significant difference among the front, middle and rear segments. As shown in Figure 7b, the case of relative localization is vastly different, as the relative localization error accumulates along a trace. The top 50 teams have an average localization error of 3.75m, 4.90m, and 5.39m for the front, middle and rear segments, respectively. Recall that we align the initial position with the ground truth when calculating the relative localization error. Combining Figure 7a and Figure 7b, we can find that accumulated errors in tracking (approximated by relative localization errors) can be effectively mitigated by positioning and various other optimizations (§3.5). For example, the difference in localization errors between the rear and front segments is reduced from 1.64m (relative localization) to 0.11m (final localization).

4.4 Floor Estimation

The top 50 teams leveraged various models for floor estimation, and the vast majority of them achieved high accuracy. As shown in Figure 8, the floor estimation accuracy across all 50 teams is at least 97.5%. Moreover, more than half of them have achieved 100% accuracy in floor estimation.

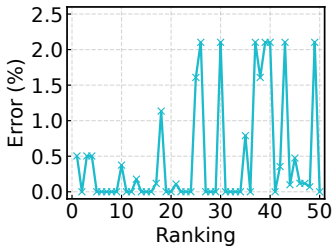


Figure 8: Floor estimation error.

In this competition, because the teams do not need to make real-time prediction and because our test set does not contain cross-floor traces, the teams can possibly use features observed in a late stage of a trajectory to determine the floor. We observe that most teams indeed adopt this strategy. For example, there may not be sufficient features to estimate the floor at the beginning of a test trace. When receiving strong WiFi signals in the middle or even towards the end of a trace, one can leverage them to estimate the floor level. Note that the aforementioned strategy is not applicable in real-world indoor localization systems where floor information is a prerequisite for localization. Therefore, we regard the high floor estimation accuracy (99.5%) as an overestimate compared to numbers reported in the literature (*e.g.*, 97% reported in [63]). To make the competition more realistic, future localization competitions should introduce more cross-floor traces or require teams to report the floor at the beginning of a test trajectory.

Table 5: Locations of test traces with floor errors.

Floor-error Location	Atrium	Dead End	Other
Ratio	73.1%	5.2%	21.6%

The locations where floor errors occur are shown in Table 5. We classify each floor into three types of areas, namely

atrium, dead end, and other (default areas). An atrium is a large, open multi-storied space in a large building where smartphones can receive radio signals from other floors. As a result, it has witnessed most (73.1%) of the floor estimation errors. 5.2% of the floor errors occur at dead ends, and they are mainly caused by a lack of strong radio signals.

We find that in the atrium and dead-end areas, KNN-based approaches (used by the 1st and 3rd-ranked team) do not exhibit the best accuracy for floor estimation (Figure 8), although they can achieve very high accuracy in positioning. LSTM-based methods can mitigate the floor estimation errors. Among the top 20 teams (see Table 1), the 7th, 8th, 9th, 11th, 15th and 16th ranked teams all had 100% floor accuracy, while the 13th and 17th ranked teams also achieved a floor accuracy of 99.8% and 99.9%. These teams all use LSTM-based methods for floor estimation. This result suggests that there is no single method that performs best for all localization modules (positioning, tracking, floor estimation, *etc.*).

■ **Finding 9.** *Most floor estimation errors occur in atrium areas. Unlike positioning where lightweight ML achieves the best accuracy, deep learning helps further boost the floor estimation accuracy. Future competitions should introduce more cross-floor traces or require teams to report the floor at the beginning of a test trajectory.*

5 DISCUSSIONS AND LIMITATIONS

Through the entire process of organizing this competition, we have obtained rich experiences and noticed several limitations of our competition as summarized below. We hope they will help future organizers of similar events.

• As described in §3.1, the localization logic of each team is usually not directly provided by the team, but inferred by us from public information. We do not require submitting source code because teams use heterogeneous tools, environments, and languages. Even if the source code is available, understanding their logic requires considerable human effort given the scale of our competition. Therefore, we take the approach of quantitatively characterizing the localization results, whereas qualitatively studying the teams' localization logic by leveraging multiple sources.

• In our competition, each team only needs to make offline estimations for waypoints in a test trace and submit the answers. This makes the grading system very easy to implement and the competition rules easy to follow. It also helps scale up the competition. However, it differs from typical real-world indoor localization systems that provide users with real-time location results. Future organizers may consider making their competitions more realistic by adding more real-time components, while keeping the rules simple and the grading overhead low.

• In our competition, a solution is evaluated purely based on its accuracy. However, other factors such as localization delay, energy consumption, infrastructure cost, and usability, to name a few, are also important. Future competitions may consider adding additional components to the grading metric. More research is needed on deriving good and fair metrics that can reflect users' true experience – this will benefit the design of real-world localization systems as well. Also, we do make several findings with implications on resource usage, energy efficiency, *etc.* For example, top solutions do not necessarily use deep learning that is heavy-weight in terms of resource usage (§3.3).

In addition, despite using accuracy as the primary metric, the key goals of our study include decomposing modern localization solutions from diverse designers, identifying their common design patterns, and finding common pitfalls. These lessons are beneficial for future indoor localization systems.

• In our competition, the top 18 teams achieve an average accuracy of no more than 3.7m. Even for the 500th team, its mean positioning error can reach as low as 5.7m (§1). One may argue that for real-world indoor localization use cases, there is not much difference between the above results. However, high tail errors exhibited by state-of-the-art localization solutions (e.g., commercial systems such as [63] and top teams in our competition) may still impact users' experience. Therefore, maintaining low errors (in particular in challenging scenarios) remains challenging and important.

• Our competition misses cross-floor test cases, making it easier for floor estimation. Also, in our dataset, landmarks have a higher probability of being an answer because, for their convenience, surveyors oftentimes leverage visually recognizable landmarks as waypoints during data collection. This lowers the bar for participating teams to "guess" the answer. Future organizers should pay attention to these details.

• The organizers should be aware of possible "side channels" that the teams may exploit. For some side channels that real-world systems can also use (*e.g.,* device fingerprinting, §3.5), organizers may consider providing the data directly, so that the teams do not need to reverse engineer the dataset.

6 RELATED WORK

Competitions. Microsoft held the very first indoor localization competition in 2014 [34], which took place in two rooms and a hallway of a hotel in Berlin. Competitors could deploy their own devices, including WiFi APs that support channel state information (CSI) [19], ultrasonic chirps, LED, and other signal transmitters. Most teams adopted traditional positioning algorithms (*e.g.,* fingerprinting, AoA [38]), and 2 teams leveraged neural network models to achieve indoor localization. In the next three years, Microsoft continued to

hold the competition in different locations [33]. IPIN [24] held its indoor localization competition [25] since 2014, consisting of both onsite and offsite tracks. In addition to radio signals and IMU data, they also provided images, 5G data and channel information for different competition tracks.

Our competition focused on infrastructure-free localization. The dataset was collected from real-world sites with off-the-shelf smartphones. Compared to Microsoft and IPIN competitions, our competition involved large-scale evaluation, of which the test set consisted of 625 traces in 24 sites. Most of the top 20 teams in our competition used neural network models to predict the locations of targets. Besides, the implementation time was reduced from several years to less than four months.

Sensor Data for Indoor Localization. As the accuracy of GPS drops significantly in indoor scenarios, researchers leverage many other data sources to facilitate indoor localization. SmartPDR [26] tracks the target by only using the IMU data. As PDR performance improves [39, 50], many works combine IMU data with radio signals (*e.g.,* WiFi) for localization. The prevalence of smartphones makes WiFi RSSI [6, 16, 37, 54] and its physical-layer information [28, 43, 60] popular in positioning. Compared to WiFi, Ultra Wideband (UWB) has higher bandwidth, and is promising to achieve lower positioning errors [18, 42, 47, 55]. Bluetooth Low Energy (BLE)-based system [5, 9, 12, 46] can avoid privacy issues that occur in WiFi scanning. BLE beacons can be powered by batteries, making them available for large-scale deployments. Besides, optical [29, 66], acoustic [48, 64], and magnetic [45, 57] data is also exploited in indoor localization.

Localization Algorithms. A variety of positioning algorithms have been proposed. Smartphones can scan the ambience fingerprint (*e.g.,* radio and acoustic signals), and many researchers leverage fingerprinting-based techniques [2, 12, 20, 48, 57, 62] to estimate the target's location. Chintalapudi *et al.* [6] use crowdsourcing to collect indoor signatures without knowing floor plans or AP locations. Since WiFi APs are common in indoor environments, fingerprinting-based indoor positioning systems have been deployed in the real world at a large scale. Yang *et al.* [61] use the difference in the arrival time of acoustic signals to locate the smartphone in the car. Vision-based solutions leverage the anchors (including its size and angle) in the image to locate the target [11, 53]. Some WiFi APs support providing the developer with CSI [19]. Therefore, researchers make use of Angle of Arrival (AoA) to locate the target [28, 58]. SAIL [36], ToneTrack [59], and Chronos [52] leverage Time of Flight (ToF) to achieve sub-meter localization accuracy. PinLoc [44] regards CSI as unique features for fingerprinting. In general, CSI-based methods achieve much higher localization accuracy but require physical layer information, which most commercial WiFi APs cannot provide.

Commercial Localization Systems. Recently, several localization systems have registered impressive large-scale commercial deployment. For example, IODetector [68] utilizes BLE beacons to track couriers and schedule on-demand food deliveries; MLoc [63] combines BLE beacons and geomagnetic fields to provide localization services for shopping malls; Tencent [37] built a Wi-Fi based indoor localization system that leverages crowdsourcing to collect fingerprints. Our paper has a different focus and complements the above works: instead of characterizing a single commercial localization system, we focus on the breadth of solutions and aim at understanding the common design patterns and pitfalls of localization solutions built by top researchers and engineers.

Survey of Localization Techniques. Researchers have also published literature survey papers on localization techniques [35, 56, 65]. Our work differs from these survey papers in several aspects. First, we organize the whole localization competition and contribute the dataset (with ground truth) to the community. Second, we conduct strategic data analysis to a large number of solutions and their results. Third, we contribute new findings and lessons instead of summarizing existing published works.

7 CONCLUDING REMARKS

We organized to our knowledge the largest open-to-public indoor localization competition in terms of the data size, the number of teams, and the number of received submissions. The main purpose of organizing this competition was to advance the field and to foster the indoor localization community. We have described the lessons learned from analyzing the submissions and experiences in organizing the competition. Overall, we were impressed by the technical merit and engineering efforts of many of the 1,170 teams with diverse background and expertise. The qualitative analysis of their localization algorithms and quantitative characterization of the competition results help significantly advance our understanding of fingerprint-based indoor localization. We believe many of the lessons and experiences are not limited to indoor localization competitions; they could potentially be applied to other competitions in mobile computing, which play a key role in developing and nurturing our research community.

ACKNOWLEDGMENTS

We thank the anonymous reviewers and our shepherd for their insightful comments. Zhimeng Yin's research was supported by NSF China 62102332, ECS CityU 21216822, and City University of Hong Kong 9610491. Jie Liu's research is partly supported by the National Key R&D Program of China under Grant 2021ZD0110905, and an Open Competition Program of Heilongjiang Province under Grant 2021ZXJ05A03.

REFERENCES

[1] Naomi S Altman. 1992. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician* 46, 3 (1992), 175–185.

[2] Martin Azizyan, Ionut Constandache, and Romit Roy Choudhury. 2009. Surroundsense: mobile phone localization via ambience fingerprinting. In *Proceedings of the 15th annual international conference on Mobile computing and networking*. 261–272.

[3] Beaconstac. 2021. 10 Airports Using Beacons to Take Passenger Experience to the Next Level. https://blog.beaconstac.com/2016/03/10-airports-using-beacons-to-take-passenger-experience-to-the-next-level/.

[4] Avrim Blum and Moritz Hardt. 2015. The ladder: A reliable leaderboard for machine learning competitions. In *International Conference on Machine Learning*. PMLR, 1006–1014.

[5] Dongyao Chen, Kang G Shin, Yurong Jiang, and Kyu-Han Kim. 2017. Locating and tracking ble beacons with smartphones. In *Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies*. 263–275.

[6] Krishna Chintalapudi, Anand Padmanabha Iyer, and Venkata N Padmanabhan. 2010. Indoor localization without the pain. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*. 173–184.

[7] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. In *SSST@EMNLP*. Association for Computational Linguistics, 103–111.

[8] Charles K Chui, Guanrong Chen, et al. 2017. *Kalman filtering*. Springer.

[9] Yi Ding, Ling Liu, Yu Yang, Yunhuai Liu, Desheng Zhang, and Tian He. 2021. From conception to retirement: a lifetime story of a 3-year-old wireless beacon system in the wild. *IEEE/ACM Transactions on Networking* 30, 1 (2021), 47–61.

[10] Petar M Djuric, Jayesh H Kotecha, Jianqui Zhang, Yufei Huang, Tadesse Ghirmai, Mónica F Bugallo, and Joaquin Miguez. 2003. Particle filtering. *IEEE signal processing magazine* 20, 5 (2003), 19–38.

[11] Jiang Dong, Yu Xiao, Marius Noreikis, Zhonghong Ou, and Antti Ylä-Jääski. 2015. iMoon: Using smartphones for image-based indoor navigation. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. 85–97.

[12] Ramsey Faragher and Robert Harle. 2015. Location fingerprinting with bluetooth low energy beacons. *IEEE journal on Selected Areas in Communications* 33, 11 (2015), 2418–2428.

[13] Christopher C Foster and Gabriel Hugh Elkaim. 2008. Extension of a two-step calibration methodology to include nonorthogonal sensor axes. *IEEE Trans. Aerospace Electron. Systems* 44, 3 (2008), 1070–1078.

[14] Google. 2022. Google Indoor Maps. https://www.google.com/maps/about/partners/indoormaps/.

[15] Google. 2022. WiFi scanning overview. https://developer.android.com/guide/topics/connectivity/wifi-scan.

[16] Abhishek Goswami, Luis E Ortiz, and Samir R Das. 2011. WiGEM: A learning-based approach for indoor localization. In *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*. 1–12.

[17] Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks* 18, 5-6 (2005), 602–610.

[18] Ismail Guvenc, Chia-Chin Chong, and Fujio Watanabe. 2007. NLOS identification and mitigation for UWB localization systems. In *2007 IEEE Wireless Communications and Networking Conference*. IEEE, 1571–1576.

[19] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. 2011. Tool release: Gathering 802.11 n traces with channel state information. *ACM SIGCOMM computer communication review* 41, 1 (2011), 53–53.

[20] Suining He, Tianyang Hu, and S-H Gary Chan. 2015. Contour-based trilateration for indoor fingerprinting localization. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. 225–238.

[21] Minh Tu Hoang, Brosnan Yuen, Xiaodai Dong, Tao Lu, Robert Westendorp, and Kishore Reddy. 2019. Recurrent neural networks for accurate RSSI indoor localization. *IEEE Internet of Things Journal* 6, 6 (2019), 10639–10651.

[22] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[23] IndoorAtlas. 2016. Hyper-local maps and navigation for millions. https://www.indooratlas.com/case/improving-indoor-mapping-capabilities-with-yahoo-japan/.

[24] IPIN. 2022. International Conference on Indoor Positioning and Indoor Navigation. https://ipin-conference.org/.

[25] IPIN. 2022. IPIN competition. https://evaal.aaloa.org/.

[26] Wonho Kang and Youngnam Han. 2014. SmartPDR: Smartphone-based pedestrian dead reckoning for indoor localization. *IEEE Sensors journal* 15, 5 (2014), 2906–2916.

[27] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems* 30 (2017).

[28] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. 2015. Spotfi: Decimeter level localization using wifi. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*. 269–282.

[29] Ye-Sheng Kuo, Pat Pannuto, Ko-Jen Hsiao, and Prabal Dutta. 2014. Luxapose: Indoor positioning with mobile phones and visible light. In *Proceedings of the 20th annual international conference on Mobile computing and networking*. 447–458.

[30] Yann LeCun, Yoshua Bengio, et al. 1995. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks* 3361, 10 (1995), 1995.

[31] Dong Li, Baoxian Zhang, and Cheng Li. 2015. A feature-scaling-based k-nearest neighbor algorithm for indoor positioning systems. *IEEE Internet of Things Journal* 3, 4 (2015), 590–597.

[32] Locatify. 2020. BLE Beacon Museum Guide with real-time user location. https://locatify.com/blog/case-studies/eldheimar-museum/.

[33] Dimitrios Lymberopoulos and Jie Liu. 2017. The microsoft indoor localization competition: Experiences and lessons learned. *IEEE Signal Processing Magazine* 34, 5 (2017), 125–140.

[34] Dimitrios Lymberopoulos, Jie Liu, Xue Yang, Romit Roy Choudhury, Vlado Handziski, and Souvik Sen. 2015. A realistic evaluation and comparison of indoor location technologies: experiences and lessons learned. In *IPSN*. ACM, New York, NY, USA, 178–189.

[35] Yongsen Ma, Gang Zhou, and Shuangquan Wang. 2019. WiFi sensing with channel state information: A survey. *ACM Computing Surveys (CSUR)* 52, 3 (2019), 1–36.

[36] Alex T Mariakakis, Souvik Sen, Jeongkeun Lee, and Kyu-Han Kim. 2014. Sail: Single access point-based indoor localization. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*. 315–328.

[37] Jiazhi Ni, Fusang Zhang, Jie Xiong, Qiang Huang, Zhaoxin Chang, Junqi Ma, BinBin Xie, Pengsen Wang, Guangyu Bian, Xin Li, et al. 2022. Experience: Pushing indoor localization from laboratory to the wild. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*. 147–157.

[38] Dragos Niculescu and Badri Nath. 2003. Ad hoc positioning system (APS) using AOA. In *IEEE INFOCOM 2003. Twenty-second Annual Joint*

Conference of the IEEE Computer and Communications Societies (IEEE Cat. No. 03CH37428), Vol. 3. Ieee, 1734–1743.

[39] Jiuchao Qian, Jiabin Ma, Rendong Ying, Peilin Liu, and Ling Pei. 2013. An improved indoor localization method using smartphone inertial sensors. In *International Conference on Indoor Positioning and Indoor Navigation*. IEEE, 1–7.

[40] Frank Rosenblatt. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review* 65, 6 (1958), 386.

[41] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1985. *Learning internal representations by error propagation*. Technical Report. California Univ San Diego La Jolla Inst for Cognitive Science.

[42] Zafer Sahinoglu, Sinan Gezici, and Ismail Guvenc. 2008. Ultra-wideband positioning systems. *Cambridge, New York* (2008).

[43] Souvik Sen, Jeongkeun Lee, Kyu-Han Kim, and Paul Congdon. 2013. Avoiding multipath to revive inbuilding WiFi localization. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. 249–262.

[44] Souvik Sen, Božidar Radunovic, Romit Roy Choudhury, and Tom Minka. 2012. You are facing the Mona Lisa: Spot localization using PHY layer information. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*. 183–196.

[45] Yuanchao Shu, Kang G Shin, Tian He, and Jiming Chen. 2015. Last-mile navigation using smartphones. In *Proceedings of the 21st annual international conference on mobile computing and networking*. 512–524.

[46] Mihail L Sichitiu and Vaidyanathan Ramadurai. 2004. Localization of wireless sensor networks with a mobile beacon. In *2004 IEEE international conference on mobile Ad-hoc and sensor systems (IEEE Cat. No. 04EX975)*. IEEE, 174–183.

[47] Lorenzo Taponecco, Antonio Alberto D'Amico, and Umberto Mengali. 2011. Joint TOA and AOA estimation for UWB localization applications. *IEEE Transactions on Wireless Communications* 10, 7 (2011), 2207–2217.

[48] Stephen P Tarzia, Peter A Dinda, Robert P Dick, and Gokhan Memik. 2011. Indoor localization without infrastructure using the acoustic background spectrum. In *Proceedings of the 9th international conference on Mobile systems, applications, and services*. 155–168.

[49] Arvind Thiagarajan, Lenin Ravindranath, Hari Balakrishnan, Samuel Madden, and Lewis Girod. 2011. Accurate,{Low-Energy} Trajectory Mapping for Mobile Devices. In *8th USENIX Symposium on Networked Systems Design and Implementation (NSDI 11)*.

[50] Qinglin Tian, Zoran Salcic, I Kevin, Kai Wang, and Yun Pan. 2015. A multi-mode dead reckoning system for pedestrian tracking using smartphones. *IEEE Sensors Journal* 16, 7 (2015), 2079–2093.

[51] Aäron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W Senior, and Koray Kavukcuoglu. 2016. WaveNet: A generative model for raw audio. *SSW* 125 (2016), 2.

[52] Deepak Vasisht, Swarun Kumar, and Dina Katabi. 2016. {Decimeter-Level} Localization with a Single {WiFi} Access Point. In *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. 165–178.

[53] Martin Werner, Moritz Kessel, and Chadly Marouane. 2011. Indoor positioning using smartphone camera. In *2011 international conference on indoor positioning and indoor navigation*. IEEE, 1–6.

[54] Chenshu Wu, Zheng Yang, Yunhao Liu, and Wei Xi. 2012. WILL: Wireless indoor localization without site survey. *IEEE Transactions on Parallel and Distributed systems* 24, 4 (2012), 839–848.

[55] Henk Wymeersch, Stefano Maranò, Wesley M Gifford, and Moe Z Win. 2012. A machine learning approach to ranging error mitigation for UWB localization. *IEEE transactions on communications* 60, 6 (2012), 1719–1728.

[56] Jiang Xiao, Zimu Zhou, Youwen Yi, and Lionel M Ni. 2016. A survey on wireless indoor localization from the device perspective. *ACM Computing Surveys (CSUR)* 49, 2 (2016), 1–31.

[57] Hongwei Xie, Tao Gu, Xianping Tao, Haibo Ye, and Jian Lv. 2014. MaLoc: A practical magnetic fingerprinting approach to indoor localization using smartphones. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 243–253.

[58] Jie Xiong and Kyle Jamieson. 2013. {ArrayTrack}: A {Fine-Grained} Indoor Location System. In *10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13)*. 71–84.

[59] Jie Xiong, Karthikeyan Sundaresan, and Kyle Jamieson. 2015. Tonetrack: Leveraging frequency-agile radios for time-based indoor wireless localization. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. 537–549.

[60] Chouchang Yang and Huai-Rong Shao. 2015. WiFi-based indoor positioning. *IEEE Communications Magazine* 53, 3 (2015), 150–157.

[61] Jie Yang, Simon Sidhom, Gayathri Chandrasekaran, Tam Vu, Hongbo Liu, Nicolae Cecan, Yingying Chen, Marco Gruteser, and Richard P Martin. 2011. Detecting driver phone use leveraging car speakers. In *Proceedings of the 17th annual international conference on Mobile computing and networking*. 97–108.

[62] Zheng Yang, Chenshu Wu, and Yunhao Liu. 2012. Locating in fingerprint space: Wireless indoor localization with little human intervention. In *Proceedings of the 18th annual international conference on Mobile computing and networking*. 269–280.

[63] Hu Yuming, Qian Feng, Yin Zhimeng, Li Zhenhua, Ji Zhe, Xu Qiang, Han Yeqiang, and Jiang Wei. 2022. Experience: Practical Indoor Localization for Malls. In *MobiCom*. ACM, 1–12.

[64] Sangki Yun, Yi-Chao Chen, Huihuang Zheng, Lili Qiu, and Wenguang Mao. 2017. Strata: Fine-grained acoustic-based device-free tracking. In *Proceedings of the 15th annual international conference on mobile systems, applications, and services*. 15–28.

[65] Faheem Zafari, Athanasios Gkelias, and Kin K Leung. 2019. A survey of indoor localization systems and technologies. *IEEE Communications Surveys & Tutorials* 21, 3 (2019), 2568–2599.

[66] Chi Zhang and Xinyu Zhang. 2016. LiTell: Robust indoor localization using unmodified light fixtures. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. 230–242.

[67] Min-Ling Zhang and Zhi-Hua Zhou. 2007. ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognit.* 40, 7 (2007), 2038–2048.

[68] Pengfei Zhou, Yi Ding, Yang Li, Mo Li, Guobin Shen, and Tian He. 2022. Experience: Adopting indoor outdoor detection in on-demand food delivery business. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*. 94–105.