This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3105188, IEEE Access

Digital Object Identifier 10.1109/ACCESS.2021.DOI

Indoor Positioning System using Synthetic Training and Data Fusion

YURI ASSAYAG[1], HORÁCIO OLIVEIRA[1], EDUARDO SOUTO[1], RAIMUNDO BARRETO[1], AND RICHARD PAZZI[2]
[1]Institute of Computing, Federal University of Amazonas, Manaus 69067-005, Amazonas, Brazil
[2]Faculty of Business and Information Technology, Ontario Tech University (UOIT), Oshawa, ON L1H 7K4, Canada
Corresponding author: Yuri Assayag (e-mail: yfa@icomp.ufam.edu.br)

This research was partially funded by the Coordination for the Improvement of Higher Education Personnel (CAPES), Institute for Innovation, Research, and Scientific and Technological Development of Amazonas (IPDEC), Positivo Technologies and, according to Article 48 of Decree no 6.008/2006, Samsung Electronics of Amazonia Ltda, under the terms of Federal Law no 8.387/1991, through agreement no 003, signed with ICOMP/UFAM.

ABSTRACT Indoor Positioning Systems (IPSs) are used to estimate the position of mobile devices in indoor environments. Fingerprinting is the most used technique because of its higher accuracy. However, this technique requires a labor-intensive training phase that measures the Received Signal Strength Indicator (RSSI) at all Reference Points (RPs) locations. On the other hand, model-based IPSs use signal propagation models to estimate distances from RSSI. Thus, they do not require expensive training but result in higher positioning errors. In this work, we propose SynTra-IPS (Synthetic Training Indoor Positioning System), a hybrid approach between a fingerprint and a model-based IPS that uses synthetic, simulated datasets combined with data fusion techniques to eliminate the fingerprint collection cost. In our solution, we use the map of the scenario, with known anchor nodes' positions and the log-distance signal propagation model, to generate several synthetic, model-based, fingerprint training datasets. In the online phase of our solution, the positions estimated by the several synthetic datasets using K-Nearest Neighbors (KNN) are combined using data fusion techniques into a single, more accurate position. We evaluated the performance of our SynTra solution in a real-world, large-scale environment using mobile devices with Bluetooth Low Energy (BLE) technology, and we compared our solution to classic approaches from the literature. Our results show that SynTra can locate mobile devices with an average error of only 2.36 m while requiring no real-world environment training.

INDEX TERMS Indoor localization, fingerprint, path-loss propagation model, synthetic dataset.

I. INTRODUCTION

Positioning systems can be defined as the process of finding the position of a target in outdoor or indoor environments [1]. Today, one of the most known positioning systems is the Global Navigation Satellite System (GNSS), which includes the Global Positioning System (GPS), that is able to locate devices in outdoor environments, where there is a line-of-sight among the device and the satellites. On the other hand, Indoor Positioning Systems (IPSs) focus on locating mobile devices in indoor environments, where GNSS can not provide a good accuracy [2]. Currently, there is a lot of research to propose new methods and technologies that increase the accuracy of the IPSs, motivated by the high complexity of indoor environments [3]–[6].

The main technology used in IPSs is based on local radio signals, and the position can be estimated using the Time of Arrival (ToA) [7], Time Difference of Arrival (TDoA) [8], Angle of Arrival (AoA) [9], and the Received Signal Strength Indicator (RSSI) [3]. The RSSI being the most frequently used due to its high availability since most devices with wireless communication, such as Bluetooth Low Energy (BLE) or WiFi, already comes with this feature. WiFi is a wireless communication technology widely available in different places such as malls and airports, which means no additional hardware and deployment requirements for indoor localization. On the other hand, BLE has also been widely used in indoor localization due to its low power consumption, allowing it to be used by energy-constrained devices such as smartwatches while also being available in most smartphones.

Most IPSs can be classified into model-based and fingerprint-based. Model-based IPSs estimate the positions

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3105188, IEEE Access

based on the distance between the mobile device and the Anchor Nodes (ANs), which are fixed devices with known positions [10]. For this, the RSSI values are converted into distances using a path-loss signal propagation model, the most known being the Log-Distance model. Then, the position computation is done using, for instance, the least-squares technique. However, due to the high complexity of indoor environments and the high RSSI variation, this conversion is not always done realistically [11].

Fingerprint-based IPSs [12] are known to be more accurate and popular. This method is divided into two phases: offline and online. In the offline phase, also known as training, several evenly spaced Reference Points (RPs) are distributed in the environment. For each RP, several RSSI values between a mobile device and the anchor nodes need to be collected. They are then stored in a dataset along with the position where the signals were collected. In the online phase, the mobile device that we want to locate sends an advertising packet that is received by the anchor nodes that estimate the RSSIs and send them to a server. The server compares these RSSI measurements to the ones in the dataset to estimate the mobile device position. This can be done using machine learning techniques such as the K-Nearest Neighbor (KNN) [1], [13]–[15]. Although fingerprint-based IPSs are more accurate, the fingerprint collection on the offline phase is very time-consuming and laborious. Moreover, the fingerprint dataset is unable to adapt to future changes in the environment, requiring a new fingerprint collection, which makes their implementation unfeasible in large-scale locations. Thus, the main challenge of this method is how to reduce the need for a real fingerprint collection.

In this work, we propose SynTra-IPS (Synthetic Training Indoor Positioning System), a hybrid approach between a fingerprint and a model-based IPS. In the offline phase of our solution, we use a log-distance propagation model with different parameters to generate several synthetic training datasets that reflect the RSSIs in the different RPs of the environment under different propagation conditions. In the online phase, we execute the K-Nearest Neighbor (KNN) in all synthetic datasets to locate a signal from the mobile node. Then, we use data fusion techniques to combine all of the estimated positions into a single, more accurate position. To evaluate the performance of our solution, we implemented a real-world, large-scale testbed using mobile devices with Bluetooth Low Energy (BLE) technology. Our results show that SynTra can locate mobile devices with a average error of only 2.36 m. As we will show, this is a better accuracy when compared to model-based solutions, getting close to a complete fingerprint-based solution, but without the need for any real-world, laborious training.

Our main contributions are summarized as follows:

1) Our solution uses several synthetic datasets to characterize the signal in different regions of the scenario, without the need for complex real-world data gathering from the environment.
2) We propose a new data fusion strategy that combines the positioning estimates by KNN using all synthetic datasets, into a single, more accurate position that outperforms approaches that use just a single synthetic dataset.
3) Through a large number of real-world experiments, we verify the efficiency and effectiveness of the proposed solution. Our results show that the system can achieve a competitive localization accuracy compared to state-of-the-art IPSs such as model-based IPSs, IPSs using a single synthetic dataset, and even traditional fingerprint-based IPSs with real training.

The rest of the paper is organized as follows. In the next section, we show our related work. Section III presents SynTra, our proposed IPS solution. Section IV shows our real-world testbed and experimentation methodology. In Section V, we show and discuss the results of the performance evaluation. In Section VI, we discuss the applicability and limitations of our solution. Finally, Section VII presents our conclusions and future work.

II. RELATED WORK

To estimate the position of a mobile device in an indoor environment is a complex task since the electromagnetic signal sent by devices does not have a deterministic behavior [16]. In order to make IPS more robust and accurate, many techniques and algorithms are proposed in the literature. Most solutions can be classified into model-based and fingerprint-based.

In model-based IPSs, the signals measured between the mobile device and at least three anchor nodes are used to estimate the distances. For this, signal propagation models such as the Log-distance [10] and Two-ray Ground Reflection Model (TGRM) [17] are used. In [18], the authors use the frequency diversity in the wireless channel to reduce the multipath effect on the distance estimation. Similarly, in [19], the authors propose the Optimal Multi-channel Trilateration Positioning Algorithm (OMCT) to find the global optimal parameter values and prevent the algorithm from falling into local optimum. Thus, the focus is reducing the multipath effects to increase system accuracy. In [20], it is proposed to use assistant nodes and an adaptive Kalman filter to assist and improve the distance estimation. However, the experiments did not consider the complexity of the environment, such as walls and other obstacles, which can result in lower accuracies.

In [21], the authors propose a model-based IPS that uses the k-means algorithm to separate the RSSI into three groups, where each group receives different filters that allow the propagation model to make more stable distance estimations. Sadowski *et al.* [3] compare the performance of a distance-based IPS using four dominant technologies: Wi-Fi, BLE, LoRaWAN, and ZigBee. The evaluation metrics were system accuracy and energy consumption, and the results show that Wi-Fi and BLE have advantages over other technologies. The model-based IPS mentioned above uses a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3105188, IEEE Access

fixed path-loss exponent to characterize the signal behavior in all regions of the scenario, which is not a suitable solution for large-scale scenarios. In [10], the authors have performed a smaller training to find different path-loss exponents that characterize each region of the scenario. The results show that using dynamic model parameters decreases the positioning error.

However, despite efforts to improve accuracy, indoor environments are complex, making it difficult to estimate distances only by analyzing the RSSI. Because of this, several proposed IPSs are fingerprint-based. Fingerprint-based IPS can use several machine learning algorithms to estimate the mobile device position, such as decision trees, random forest, KNN, and deep learning. In [22] the authors describe the main machine learning algorithms that can be used for localization.

A known work that uses this approach is the RADAR [23], which combined empirical measurements in the proper environment with a signal propagation model to estimate the target location. Similarly, in [12], the reduction of the empirical data needed by RADAR motivated the solution. The authors use clustering techniques to reduce computational requirements. Torres *et al.* [24] proposed a fingerprint-based IPS for home monitoring. Their solution showed that it is possible to get a precise positioning at the room level with no extra access point, an accessible solution for home monitoring.

Unlike the works that use the conventional fingerprint, crowdsourcing-based approaches use the user's movements to generate the radio map and reduce the effort to implement the system. In [25], the authors presented a location algorithm that uses a few RSSI's measured by users in real-time to update the dataset with no complete training. Similarly, Niu *et al.* [26] developed a crowdsourcing-based IPS, called WicLoc, which builds the fingerprint dataset by recording user movements, as well the RSSI, achieving room-level location accuracy. However, these systems require many sensors such as an accelerometer and gyroscope to get users' movement. Although our work is not based on crowdsourcing, our solution can be used without effort to generate immediate results through the synthetic dataset, and crowdsourcing can be used to improve the result based on the real users' data.

Other solutions use a virtual dataset generated by mathematical models to reduce the training effort. In [27], it is proposed a method to create the dataset in real-time using an optimized ray-tracing algorithm. Similarly, the authors in [28] proposed a new method for interpolating the RSSI using a path-loss model containing wall attenuation. However, these methods require the material type of the walls, which is hard to get. In [29], the authors use a deep neural network to reduce the radio map generation workload by learning the data distribution. Similarly, Kim *et al.* [30] propose a new architecture to reduce the dimension of the resource space and thus reconstruct the radio map using a deep neural network. However, training a neural network requires a lot of labeled, trained data.

Ali *et al.* [31] explores the floor plan and wall map of the environment to assistant the signal propagation model and generate the simulated training base. The experimental results show that by using the floor plan information and environmental parameters, it is possible to achieve significant positioning accuracy. In [15], it is proposed a method that requires only a few reference points to reconstruct a denser training dataset. The method uses a signal propagation model based on zone and interpolation to generate the RSSI. In [32], the proposed solution requires little training to learn the model parameters and then generates extra RSSI values in new, virtual RPs. With the common goal of reducing training workload, in [33], the authors introduce the Hierarchical Positioning Algorithm (HPA). This algorithm creates several sub-dataset with different densities in virtual RPs. However, the author uses only a sufficiently small number of fingerprints, with the same path-loss exponent in all RPs.

In Table 1, we show the comparison between the main works mentioned in this section. The positioning error of the mentioned works depend on the way the experiment was carried out (real or simulation), as well as the size of the scenario and the algorithms used to estimate the positioning.

TABLE 1. Comparison of different indoor positioning systems.

Solution	*Type*	*Training*	*Obstacle Count*	Data Fusion
[28]	fingerprint model-based	synthetic	no	no
[33]	fingerprint-based	synthetic	yes	no
[20]	fingerprint model-based	real	no	no
[23]	fingerprint-based	real	no	no
[26]	crowdsourcing	real	no	no
[31]	fingerprint-based	synthetic	yes	no
[10]	model-based	real	no	no
SynTra	fingerprint model-based	synthetic	yes	yes

Our proposed approach differs from all of the above solutions. First, differently from model-based solutions that convert real RSSIs into estimated distances, our solution converts real distances from the map into synthetic RSSIs, which allows it to take into consideration the walls of the scenario, among other things. When compared to fingerprint-based IPSs, most solutions either try to reduce the training using sensors, data analysis, and crowdsourcing or try to reduce the dataset to improve performance. Our solution completely eliminates the real-world training part of the fingerprint technique and replaces it with synthetic datasets. In particular, in the more directly related works [33] and [31], the authors present calibration-free positioning techniques, which exploit the floor plan/wall map of the environment for the construction of RSSI maps, calculating the path-loss of the signals using a signal propagation model. However, in this case, the authors generate only a single synthetic dataset to represent the signal behavior in the environment, with the same path-loss exponent in all RPs.

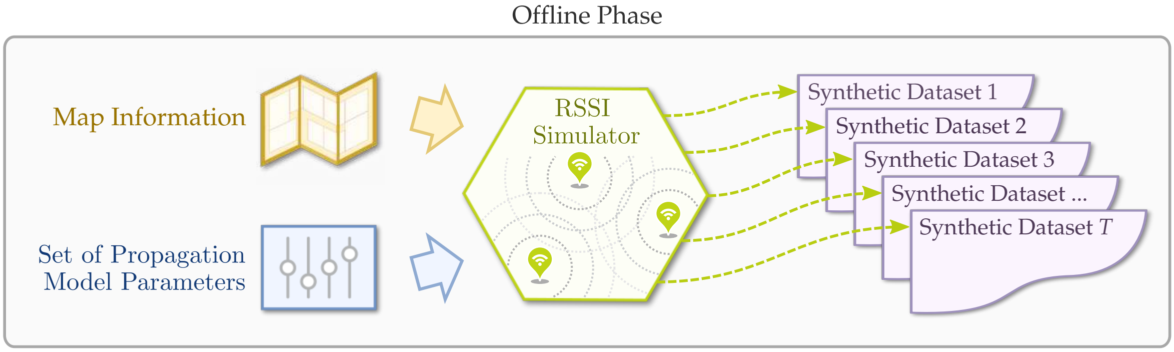

FIGURE 1. Offline phase of our SynTra architecture: map information as well as a set of propagation model parameters are used as an input to a log-distance-based RSSI Simulator that outputs a number of synthetic training datasets.

In our work, we use several synthetic datasets combined using the proposed data fusion techniques to improve the accuracy.

To the best of our knowledge, no existing work considered exploiting the log-distance signal propagation model to generate several synthetic fingerprint datasets. In addition, this article is the first to apply data fusion from several datasets to provide an improved position estimation in indoor localization. The details of our proposed solution are described in the next section.

III. SYNTRA-IPS ARCHITECTURE

In this section, we present our proposed SynTra-IPS (Synthetic Training Indoor Positioning System). Like most IPS solutions, SynTra is composed of two phases: offline and online. In the next sections, we present the details of both phases.

A. OFFLINE PHASE: DATASETS CONSTRUCTION

Figure 1 shows an overview of the offline phase. In this phase, we use the map information and an RSSI simulator to obtain several synthetic training datasets generated by a log-distance model with different propagation parameters. Thus, some datasets will eventually be better to characterize the scenario than others.

In this phase, we assume that an area A contains a set of n Anchor Nodes (ANs) with previously known positions, and then we measure the signal strength at m Reference Points (RPs) for their neighboring ANs. The RPs are evenly separated, and their positions $(X_i,\ Y_i)$, $i\ =\ (1,2,3,...,m)$ are also known. Thus, for each RP, we have vectors of received signal strengths defined as: RP_i=$(RSSI_1, RSSI_2, RSSI_3, ..., RSSI_m, Label_i, X_i, Y_i)$, where RSSI$_m$ is the received signal strength from the m^{th} AN, Label$_i$ is the RP$_i$ identification, and (X_i, Y_i) is the RP position.

A fingerprint dataset is composed of several measurements of signals at the RPs, and we associate the signals with their real locations. Usually, this dataset is created based on a real-world training step to collect the signals at each RP. However, as mentioned, this is an intensive laborious step, especially in medium to large-scale scenarios, since it requires several days to collect all of the data. Also, there is a need to re-create a new dataset when some scenario characteristics change.

Thus, to eliminate the cost of collecting fingerprints, we propose an RSSI simulator to create synthetic fingerprint datasets based on map information and virtual reference points that match the distribution of real RPs. Our goal is to get the information of the scenario through the floor plan of the building and, thus, reduce the hard step to create the signal map. The RSSI simulator was developed by our research group. As input, it requires the real ANs' positions, the location of the RPs, and the location and dimensions of the rooms' walls. As an output, our RSSI simulator generates a set of synthetic datasets using different parameters for the signal propagation model, in our case, the log-distance [34], [35]. These signals, for each RP, can be computed as:

$$RSSI_n = RSSI_{d0} - 10\alpha \log(\frac{d}{d_0}) - \sum_i L_i + X_\sigma \quad (1)$$

where $RSSI_n$ is signal strength received from the n^{th} AN, d is the distance between the RP and AN$_n$, $RSSI_{d0}$ is the RSSI value measured at distance d_0 (usually 1 m), α is the path-loss exponent, i.e., a signal loss rate related to the environment, $\sum_i L_i$, is the attenuation constants in dB for the quantity of walls between the RP and AN$_n$ and, finally, X_σ is a zero-mean Gaussian random variable that models the RSSI variation [36]. When establishing the parameter values in (1), it is possible to get several synthetic signal values for all of the RPs, and then create a single, synthetic training dataset.

However, it is known that signal propagation in indoor environments is subject to several challenges since obstacles can cause high signal variation. Thus, different areas of the scenario may have different parameters in the propagation model that best characterizes the signal's behavior. Considering that we have four main parameters in the log-distance model ($RSSI_{d0}$, α, L_i, and X_σ), we can establish different values for each parameter and create T different

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3105188, IEEE Access

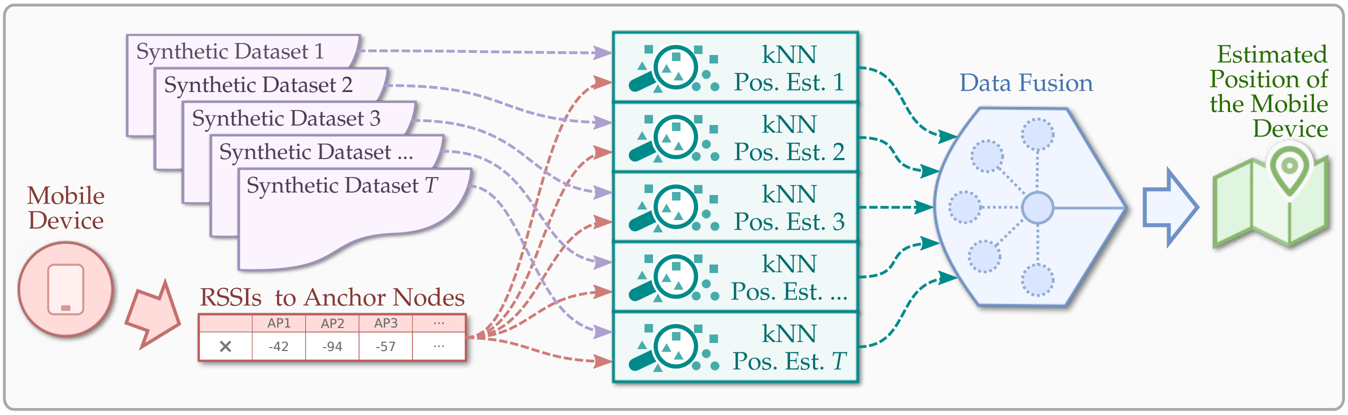

FIGURE 2. Online phase of our SynTra architecture: positions estimated by several synthetic datasets, using K-Nearest Neighbors (KNN), are combined using data fusion techniques to form a single, more accurate position.

training datasets with all values combinations that can be tuned to make it fit nearly any regions of the environment. For instance, combining parameters values of $RSSI_{d0}$ = {-55, -60}, α = {2.5, 3.0, 4.0}, L_i = {2, 3}, and X_σ = {1,2,3}, it is possible to get 36 different synthetic datasets. Some of them will perform better in different areas of the scenario.

Therefore, the result of the offline phase of our SynTra-IPS is a set of synthetic datasets with different parameter values for the propagation model. In the next section, we will show how to combine the results of these synthetic datasets to estimate the users' positions.

B. ONLINE PHASE: ESTIMATING POSITIONS

In the online phase, the RSSI values of a mobile device are used to estimate its position. Figure 2 shows an overview of the online phase of our SynTra-IPS. In this phase, the positions estimated by the several synthetic datasets, using K-Nearest Neighbors (KNN), are combined using data fusion techniques to form a single, more accurate position.

The online phase starts when a mobile device sends a packet. This packet will be received by several ANs that will be able to estimate the RSSIs. These RSSIs are sent to a central location server where the synthetic datasets, computed in the offline phase, are stored.

In the next step, for each synthetic dataset, we find the synthetic sample that best matches the real-world RSSIs samples from the mobile device. For this, several machine learning techniques can be used. In SynTra, we used the KNN algorithm, one of the most popular techniques used in fingerprint-based IPSs. KNN uses the Euclidian distance as a similarity measure to find the dataset sample that is most similar to the real-world RSSIs. Thus, the position estimation of the mobile device, using that specific synthetic dataset, is the same position as the RP from that closest sample.

After executing the KNN for each one of the T synthetic datasets, we will have T possibly different position estimations, each one with its own accuracy, depending on how close the propagation model parameters of the synthetic dataset matches the characteristics of the real-world area the mobile device is located.

Finally, the last step of the online phase is how to combine all of these T position estimations into a single, more accurate one. For this, we use data fusion techniques. Data fusion allows us to combine data from several sources in such a way that the accuracy of the resulting estimation is higher than any of the individual sources. For our SynTra solution, we proposed and evaluated the performance of four data fusion techniques, which will be explained in the next paragraphs.

1) SynTra Voting

In the first data fusion technique, we used a simple majority voting mechanism to determine the best position. Thus, we consider the final position to be the one that was most often chosen among the T predictions. Here, voting is done using the point identification label. We will refer to this variation of our solution as *SynTra Voting*.

2) SynTra Dist

In the second data fusion technique, called *SynTra Dist*, we use the euclidian distances between the matched sample and the k-nearest samples as a measurement of accuracy. Thus, for each of the T synthetic datasets, instead of having only the estimated position, we will also have this local distance information for each k-nearest samples to indicate how accurate the estimated position is.

Thus, given a position estimation and the computed distance for each one of the T synthetic datasets, we will choose the position estimation from the dataset with the lowest global distance. This approach considers that if the distance value is low, we will have more chance of choosing a synthetic dataset in which the propagation parameters more closely resembles that of the real-world region where the mobile device is located.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3105188, IEEE Access

3) SynTra Avg

In the third data fusion technique, called *SynTra Avg*, we simply get the average position (X, Y) among all of the T position estimations, as shown in (2):

$$(X, Y) = \frac{\sum_{i=0}^{T} (X_i, Y_i)}{T} \tag{2}$$

where (X, Y) is the final estimated position, (X_i, Y_i) is the position estimated using the i^{th} synthetic dataset, and T is the number of synthetic datasets. This is a simple approach that considers that, on average, the several position estimations from the T synthetic datasets are in nearby regions. However, this approach is sensitive to outliers, i.e., estimated positions with higher errors due to the unrealistic propagation parameters from their synthetic datasets.

4) SynTra WAvg

Finally, the last data fusion approach, called *SynTra WAvg*, is a combination of *SynTra Dist* and *SynTra Avg*. It tries to solve the outliers problem of *SynTra Avg* using the distance metric, used in *SynTra Dist*, as weights.

First, in order to invert the distances so the higher the better, we need to compute the sum of the weights as follows:

$$sumDist = \sum_{i=0}^{T} (maxDist - dist_i) \tag{3}$$

where $maxDist$ is the maximum distance identified among all of the T predictions, and $dist_i$ is the distance value among the k neighbors in the i^{th} synthetic dataset. Thus, the final position can be computed as follows:

$$(X, Y) = \sum_{i=0}^{T} \left(\frac{maxDist - dist_i}{sumDist}\right) * (X_i, Y_i) \tag{4}$$

Therefore, the final position is computed using the weighted average positions from all of the T estimated positions, prioritizing the ones with shorter distances, and reducing the outliers influence.

IV. EXPERIMENTAL TESTBED

This section presents our experimentation methodology and real-world testbed. The results of the performance evaluation will be discussed in Section V.

A. SYSTEM ENVIRONMENT

To evaluate the performance of SynTra, we conducted an experiment in a real, large-scale environment with an area of 645 m^2 with 15 anchor nodes distributed throughout the area. The test scenario consists of 15 spaces (11 rooms plus 3 halls), as shown in Figure 3, in which each space is covered by at least one anchor node. The anchor nodes are fixed on the ceiling in locations where it was somewhat convenient to connect them to the power supply.

Even though our solution does not require any real-world training, we still need to define reference points for the generation of the synthetic datasets. Thus, we separated the environment into 150 different reference points, evenly spaced 2 m apart from each other. Finally, combined with the floor plan information, shown in Figure 3, we have all of the required information to generate the synthetic datasets. We can then apply the signal propagation model described in (1) to simulate RSSI values at all RPs.

B. SYNTHETIC DATASET PARAMETERS

Indoor environments are complex structures hard to be modeled by a single signal propagation model since different areas have different signal characteristics caused by the diversity in layouts and obstacles that cause multipath and reflections [10], [35]. Even during the day, these signals can vary due to crowd mobility. The log-distance propagation model has parameters that require calibration to generate simulated signals that are mostly similar to real-world signal behavior.

To represent the signal in the different areas, it would need several parameter values for the log-distance model, but performing the calibration of the parameters is costly, and the effort to do so would be equivalent to performing a real collection in all RPs. Therefore, we use a range of values for the parameters, with common values to be found in IPS [3], [21], [37], [38]. In this case, we would avoid the effort of carrying out an extensive experiment to calibrate those parameters.

Instead of creating just one synthetic dataset with fixed, averaged parameters to model the whole scenario, our proposed SynTra generates several synthetic datasets with different signal parameters in such a way that eventually one of the datasets will be better than the others to represent a specific region. Thus, given the possible values that each of the parameters of the log-distance model can assume, the combination of these parameters generates several synthetic datasets. Table 2 shows several possible values for these parameters and the resulting number of possible combinations, which is the total number of synthetic datasets.

In this table, similar to (1), $RSSI_{d0}$ represents the possible values for the RSSI at 1 m, α represents the values for the path-loss exponent, L_i, the values for wall losses, and X_σ, the RSSI variations. For instance, in the first row, by

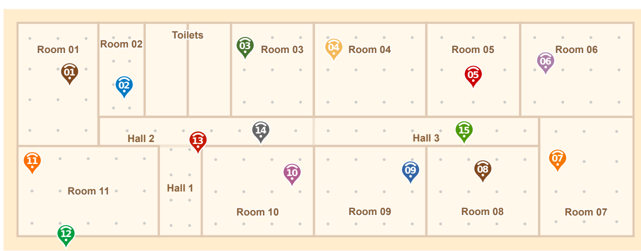

FIGURE 3. Real-world experimentation testbed: 11 rooms, 3 halls, and 15 anchor nodes. Gray dots represent the 150 reference points.

TABLE 2. Combination of synthetic datasets that can be generated by different parameters values for the log-distance model.

Model Parameters				*Possible Combinations*
$RSSI_{d0}$	α	L_i	X_σ	
55,60	3.5, 4.0, 5.5	2,3	1,2,3	36
50,55,60	3.5, 4.0, 4.5	2,3	1,2,3	52
50,55,60	3.5, 4.0, 4.5	2,3,4	1,2,3	81
50,55,60	3.5, 4.0, 4.5	2,3,4,5	1,2,3	108
50,55,60,65	3.5, 4.0, 4.5	2,3,4,5	1,2,3	240
50,55,60,65	3.5, 4.0, 4.5	2,3,4,5,6	1,2,3,4	400

combining all of the possible values for these parameters, it is possible to generate 36 different synthetic datasets. One issue with this combination is the rapid increase of the number of datasets, which results in a higher processing cost on the online phase. Using the combinations in the last row, for instance, would result in 400 different synthetic datasets.

FIGURE 4. Signal characterization of the environment using the signal propagation model as executed by the RSSI simulator.

Figure 4 shows a simulation using just one dataset generated by the following parameter values: $RSSI_{d0} = -60$, $\alpha = 4.5$, $L_i = 4\ dBm$, and $X_\sigma = 1$. In this figure, we can see how a single synthetic dataset represents the signal behavior in the scenario.

C. EXPERIMENTAL METHODOLOGY

To validate our proposed solution, we performed a real, laborious RSSI collection at the same reference points described in the previous section and depicted in Figure 3. During the experiments, the anchor nodes received BLE advertising packets sent by beacons at a 1 Hz rate. Beacon nodes are mobile devices that we will estimate the positions in the online phase and they operate with a single, small, and long-lasting battery. For the experiment, we used 11 different beacons to diversify the RSSI behavior.

Thus, at each RP, the signal values among beacons and anchor nodes are estimated and sent to a central server that then stores the data in a real-world fingerprint dataset. We collected 100 fingerprint measurements at each RP. During the experiments, the highest communication range observed between beacons and ANs was 25 m, even though at this distance, most packets are not received.

After the training process, the fingerprint dataset had 15.000 samples (signal measurements) from different beacons at 150 RPs. Again, it is important to emphasize that this is an exhaustive process and it is unnecessary for our proposed solution, being performed only for evaluation purposes to be compared to real-world data. Figure 5 depicts the average RSSI values from the real signal propagation in our test environment.

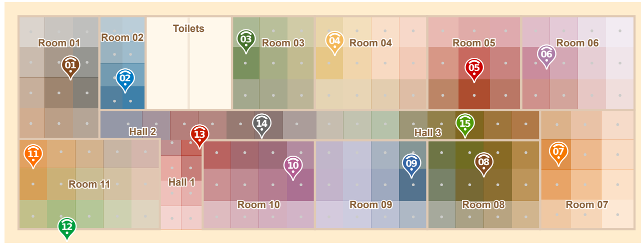

FIGURE 5. Signal characterization of the scenario based on the measurements made empirically.

V. PERFORMANCE EVALUATION

We evaluated the performance of SynTra in three different aspects. First, we analyzed the impact of the number of dataset combinations on the positioning error. Second, we evaluated the performance of the data fusion techniques. Finally, we compared the performance of our SynTra solution to traditional approaches found in the literature. In all of the experiments, we used a fixed value of 10 for the k parameter of KNN since it had the best results even though the difference from other k values was not significant.

A. DATASETS COMBINATIONS

A key aspect of our proposed SynTra solution is the number of synthetic datasets generated by combining the propagation model parameters. To evaluate the impact of the number of datasets on the system performance, we executed our solution using the different combinations of parameter values specified previously in Table 2. Thus, we executed SynTra using small combinations composed of only 36 datasets up to larger combinations of 400 datasets.

TABLE 3. Impact of the number of datasets on the average positioning error for the different data fusion approaches, highlighting the best result.

Number of Datasets	*Best Dataset*	*Data Fusion Technique*			
		Voting	*Dist*	*Avg*	*WAvg*
36	2.84 m	3.04 m	2.85 m	2.63 m	2.53 m
52	2.84 m	2.94 m	2.84 m	2.43 m	2.38 m
81	**2.84 m**	**2.93 m**	**2.85 m**	**2.42 m**	**2.36 m**
108	2.84 m	3.01 m	2.90 m	2.44 m	2.37 m
240	2.84 m	3.18 m	2.85 m	2.54 m	2.41 m
400	2.84 m	3.10 m	2.89 m	2.51 m	2.41 m

Table 3 shows the average positioning error obtained when using these different datasets combinations for all of the data fusion techniques (*Voting, Dist, Avg,* and *WAvg*) as well as the result of the *best*, single dataset. The first thing we can notice when focusing on the last column of the table is that the *WAvg* data fusion technique resulted in the smallest average error and that it was effective in reducing the error from 2.84 m (without data fusion) to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3105188, IEEE Access

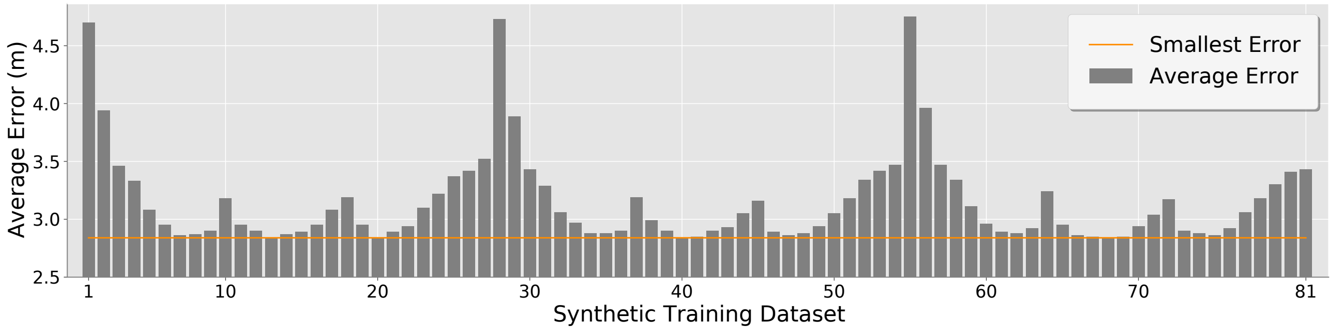

FIGURE 6. Average positioning error for each of the 81 individual, synthetic datasets; each bar corresponds to a specific dataset, i.e., a specific combination of parameters for the propagation model. The orange line highligths the smallest error among the datasets.

2.36 m. Also, we can see that the combination with 81 datasets (highlighted row) resulted in the best performance. Finally, for any combination with more than 36 datasets, the positioning error does not change significantly, ranging from 2.36 m to 2.41 m, actually increasing slightly for a higher number of datasets (e.g., 400 datasets resulted in 2.41 m).

To better understand the behavior of the positioning error for each of the individual datasets without using data fusion, Figure 6 shows the error resulted from each of the 81 synthetic datasets (highlighted row in Table 3). Each bar corresponds to a specific dataset, i.e., a specific combination of parameters for the propagation model. As we can see, the error obtained by the individual datasets can vary a lot, depending on how well the propagation model parameters represent the real-world scenario. For example, some bases have an error of almost 5 m, while others have the smallest error of 2.84 m. However, as we will see in the next section, we can reduce even more this error by using data fusion.

B. DATA FUSION TECHNIQUE

Another key aspect of our proposed SynTra solution is to combine the positions estimated by the several synthetic datasets into a single, more precise position. For this, we have proposed four different data fusion techniques: *Voting, Dist, Avg,* and *WAvg*. In this section, we evaluate their performance. For this, we used the combination of the 81 synthetic datasets highlighted previously in Table 3 with propagation model parameters detailed in Table 2.

Figure 7 shows the average error resulted when using each data fusion techniques. As we can see, *SynTra WAvg* resulted in an average error of 2.36 m, being the most accurate technique, followed by *SynTra Avg* with 2.42 m. The worst result observed was 2.93 m, from *SynTra Voting*. In the last section, we saw that by using only a single synthetic dataset, without data fusion, we could get an average error of 2.84 m in the best dataset. Thus, only *SynTra Avg* and WAvg really resulted in a better solution than any of the individual datasets, with *SynTra Dist* being very close. However, it is important to note that in a real-life application, we do not which of the individual datasets would be the best without doing the laborious real-world training.

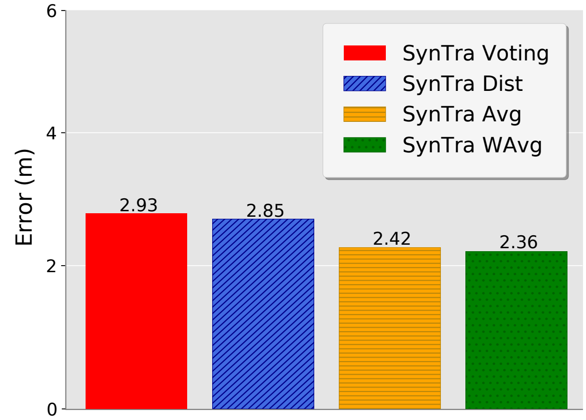

FIGURE 7. Comparison of the average positioning error for the different data fusion techniques.

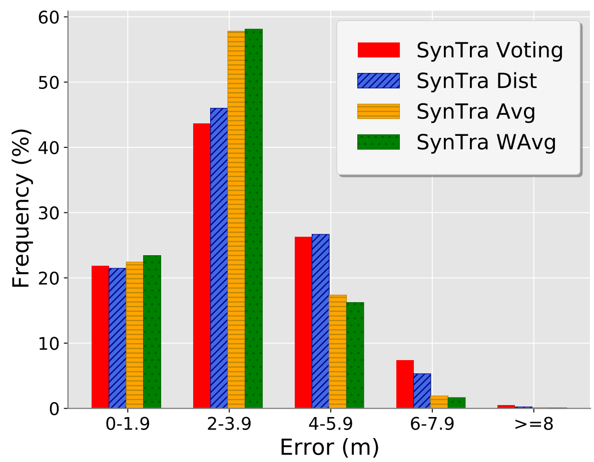

FIGURE 8. Error distribution of the estimates positions for the different data fusion techniques.

Figure 8 shows the distribution of the positioning errors. As we can see, in the case of *SynTra WAvg*, almost 60% of the errors are between 2 m and 4 m, while more than 20% are less than 2 m. Figure 9 presents the cumulative

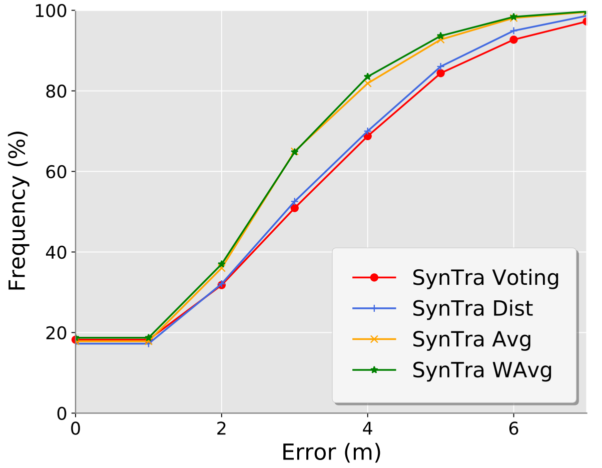

FIGURE 9. Cumulative error of the position estimations for the different data fusion techniques.

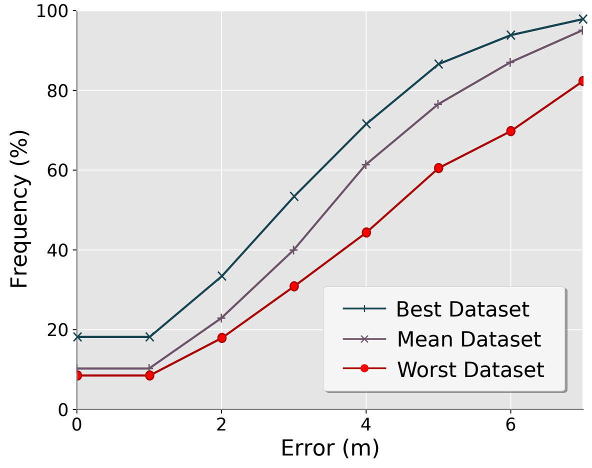

FIGURE 10. The cumulative error for the individual datasets with smallest (2.84 m), mean (3.52 m), and largest (4.75 m) positioning error.

error of the position estimations for each of the data fusion techniques. This graph shows the percentage of estimations (Y-axis) with an error smaller than the X-axis. The sharper the curve, the better since most of the estimations have smaller errors. As we can see, *SynTra WAvg* were able to achieve the lowest errors, having almost 85% of the estimations with an error smaller than $4\ m$.

The main reason *SynTra Avg* and *WAvg* resulted in the best solutions is that they use all of the 81 predictions from the synthetic datasets to compute their positions. In these solutions, the final estimated position is taken by averaging the coordinates from all predicted positions in each synthetic dataset.

In the case of *SynTra Avg*, the estimated position can be affected by outliers caused by datasets with unrealistic propagation model parameters. As can be seen in Figure 6, the accuracy is different according to the use of each offline synthetic dataset. In this figure, the orange line highlights the dataset with the smallest average error, in this case, $2.84\ m$, while other datasets resulted in a $4.75\ m$ average error. To better observe and compare the behavior of some specific datasets, Figure 10 shows the cumulative error of 3 synthetic datasets that resulted in the smallest, average, and largest positioning errors. In this figure, the best dataset is generated by the parameters $RSSI_{d0} = -55$, $\alpha = 4$, $L_i = 3\ dBm$, $X_\sigma = 3$, while the mean dataset is generated the parameters $RSSI_{d0} = -50$, $\alpha = 3.5$, $L_i = 2\ dBm$, $X_\sigma = 3$, and finally, the worst dataset is generated by the parameters $RSSI_{d0} = -60$, $\alpha = 4.5$, $L_i = 4\ dBm$, $X_\sigma = 1$. We can see that just by varying some parameter values, the average positioning error is very different. In the best dataset, more than 70% of the position estimations resulted in errors lower than $4\ m$, while in the worst dataset, only 40% of the estimations were lower than the same error.

For this reason, the *SynTra WAvg* is proposed to penalize outliers and benefit from estimates closer to the real position.

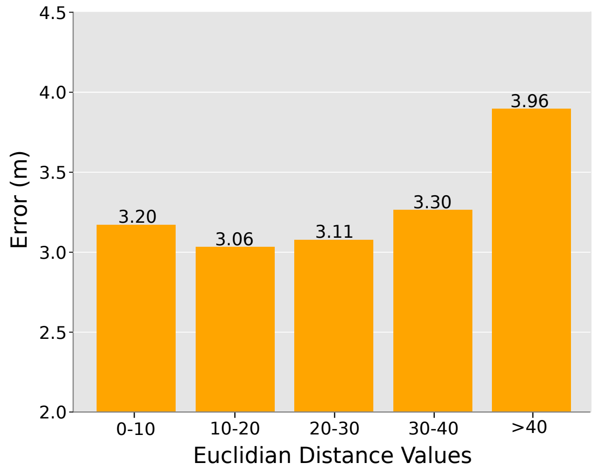

FIGURE 11. Positioning error by Euclidean Distance. Higher Euclidean distances can be used to identify higher positioning errors. Higher distance values can increase the positioning error by almost 1 m.

For this, we use a quality measurement based on the Euclidean distance from the estimated sample to the real-world sample, as explained in Sections III-B2 and III-B4. Then, we use all of the 81 predictions to estimate the final beacon position, penalizing the predictions more distant.

To better visualize how the Euclidian Distance between the estimated sample and the real-world sample can predict outliers, Figure 11 shows the average positioning error by this distance. As we can see, even though this metric is not able to indicate how small an error will be, it can indeed identify the position estimations with higher errors.

C. COMPARISON WITH OTHER SOLUTIONS

In this section, we compare the performance of our *SynTra WAvg* to traditional IPS approaches from the literature. We analyze the results provided by the different approaches in our scenario. The evaluated approaches are:

1) *Model-based:* a multilateration-based solution that uses the log-distance propagation model, as in [19], [21].
2) *Best Dataset:* a fingerprint-based IPS using our best, single synthetic dataset, similar to [31].
3) *Real Training:* a fingerprint-based IPS with a complete, laborious training of the whole area, as in [12], [23], [24].

Model-based IPSs require only minimal training to estimate an unknown position. This training is required for finding the log-distance model parameters that allow the estimation of distances between the mobile devices and the anchor nodes through the measured signal strengths. In this evaluation, we used the log-distance model with the parameters $RSSI_{d0} = -55$, and $\alpha = 4.2$. We chose these values based on the signal samples collected during our training, and we confirmed those were the best possible values, resulting in the smallest positioning errors. The positioning estimate was done using the least-squares algorithm.

For the *Best Dataset*, we carried out an experiment to find the best log-distance model parameters for a single synthetic dataset. This solution represents the one proposed in [31], as mentioned in the related work section. However, in this case, the authors generate only a single synthetic dataset to represent the signal behavior in the whole environment. To be fair in our comparisons, we consider this single dataset to be the best synthetic dataset generated by the propagation model. However, as mentioned earlier, finding such parameters remains a challenge and requires real-world training. In addition, for large-scale scenarios, this approach is not ideal to characterize the signal behavior in the different regions of the scenario.

Finally, to evaluate the performance of the traditional fingerprint using a *Real Training*, we separated our real-world data collection into training and testing, in which the measurements from 8 beacons were used to train the model, and the measurements from the other 3 beacons were used for testing. The KNN algorithm with the parameter $K = 10$ was used to find the reference point with signals most similar to those measured in the online phase. This approach can be seen as the best-case scenario since we gathered real-world RSSI data from the experimented area. The main goal of our solution is to get as close as possible to this approach but without requiring the laborious training phase.

Figure 12 shows the average positioning error of the evaluated approaches. As we can see, the average error for the model-based solution is 3.60 m, being the highest error among all approaches. The main reason for this is that the signal transformation into distance using a signal propagation model with fixed parameters is unreliable. In addition, the high RSSI variance, which is natural in indoor environments, makes this task even more complex. Hence, fingerprint-based techniques are most widely used since they result in lower positioning errors.

Still in Figure 12, we can see that the positioning error decreased when we used only one synthetic dataset generated by the best parameter values, resulting in 2.84 m. However, as mentioned earlier, although there is no need to transform the signal into the distance, in a real-world application, we would not know which of the several datasets would result in the smallest error without requiring a real-world training phase.

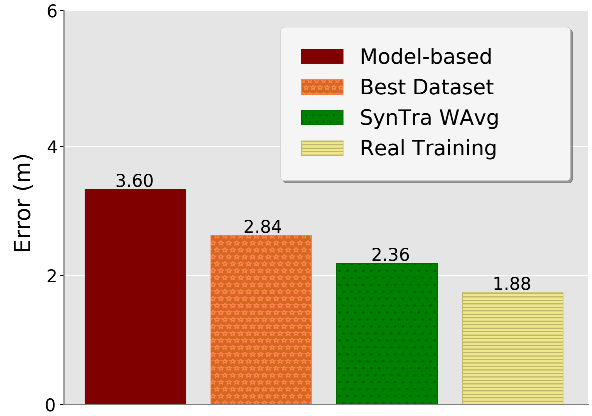

FIGURE 12. Average error of the evaluated methods: model-based IPS, fingerprint with the best, single synthetic dataset, conventional fingerprint with a real training dataset, and our proposed solution.

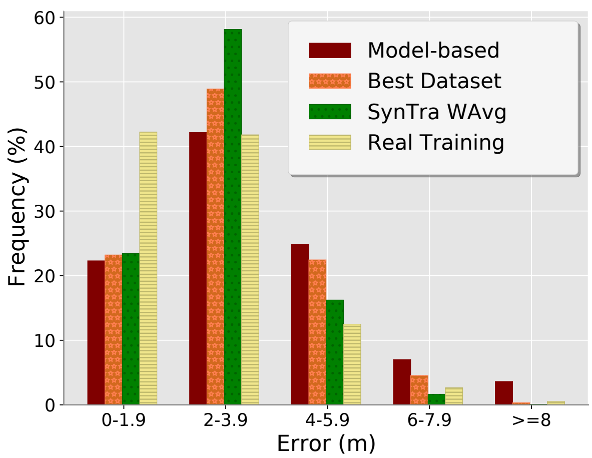

FIGURE 13. Error distribution of the evaluated IPS techniques.

Our proposed *SynTra WAvg*, resulted in better position estimations than the previous approaches with an average of 2.36 m, being almost 20% lower than the best, single synthetic dataset and 35% better than a model-based solution. Figure 13 shows that our approach contains a higher number of measurements with lower positioning errors when compared to these approaches, behind only the fingerprint with real-world training, which is the best case possible.

Finally, the fingerprint technique with a *Real Training* dataset resulted in the lowest average error among all of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3105188, IEEE Access

TABLE 4. Table with average error per room comparing the different approaches, highlighting the smallest mistakes compared to our approach.

ROOM	*SynTra Voting*	*SynTra Dist*	*SynTra Avg*	*SynTra WAvg*	*Model-based IPS*	*Best Dataset*	*Real Training*
Room 01	3.54	3.60	**2.89**	2.95	3.72	3.48	**2.70**
Room 02	3.67	3.39	3.26	**3.16**	3.90	3.21	**2.02**
Room 03	2.77	3.27	**2.33**	2.43	4.02	2.88	**2.21**
Room 04	2.68	3.34	**2.27**	2.38	3.59	2.74	2.40
Room 05	2.60	2.88	**2.14**	2.23	3.54	2.39	2.20
Room 06	3.29	3.38	2.45	**2.44**	4.29	3.10	**2.31**
Room 07	4.13	2.99	2.95	**2.68**	2.75	3.57	**2.07**
Room 08	3.14	2.80	2.28	**2.05**	3.07	3.23	**1.80**
Room 09	3.16	2.62	2.71	**2.60**	**2.60**	3.20	**2.03**
Room 10	3.74	3.43	**2.81**	2.93	3.23	3.68	**1.86**
Room 11	2.90	2.86	3.12	**2.84**	3.20	2.91	**1.62**
Hallway 1	1.49	1.36	1.31	**1.30**	4.49	1.4	**0.84**
Hallway 2	1.44	**1.40**	1.66	1.44	3.34	1.25	**0.57**
Hallway 3	**0.77**	1.19	0.87	0.92	5.69	0.91	**0.74**
Average	2.93 m	2.85 m	2.42 m	2.36 m	3.60 m	2.84 m	1.88 m

the mentioned techniques. The main reason is that this technique uses the training dataset with signal measurements from the real-world environment. Thus, despite the signal propagation model being able to adjust its parameters to generate synthetic signals similar to real-world signals, the propagation channel has complex characteristics in indoor environments. Thus, an approximation of this signal behavior is the maximum that we can achieve.

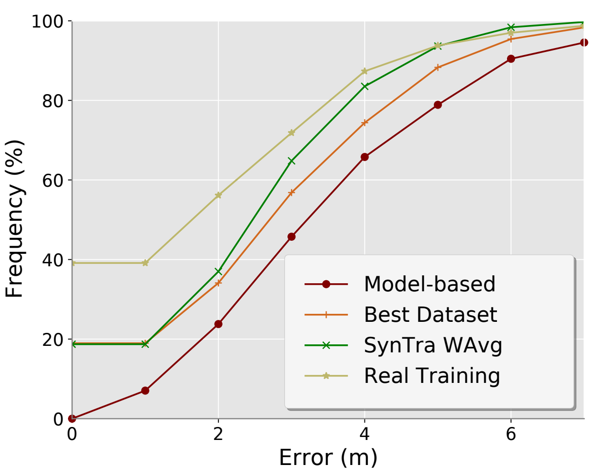

FIGURE 14. Cumulative error of the position estimations.

Figure 14 shows that the fingerprint with *Real Training* dataset has about 55% of the position estimations with an error smaller than 2 m, followed by our approach with almost 40% of the estimations. On the other hand, in our *SynTra WAvg*, more than 80% of the estimations have an error smaller than 4 m, almost the same as the fingerprint with *Real Training*. Thus, our solution was able to get close to the best-case scenario, with a difference of only 0.48 m, but without requiring any real-world training. As mentioned earlier, a possible solution to bring our solution closer to the real world would be to use crowdsourcing to supplement synthetic datasets with real data and obtain a hybrid solution.

To better understand the behavior of the errors throughout the evaluated scenario, in Table 4, we separate the average error per room obtained by each approach. To facilitate comparison, we use values in bold. In this table, we can see, as expected, that the fingerprint with real training was the one with the lowest errors per room. However, we can see that the accuracy per room varies a lot according to the approach used due to factors such as the number of reference points, anchor nodes coverage, and obstacles. Model-based IPS are the ones that result in the highest average error per room, with high errors mainly in hallways, rooms 2, 3, and 6. This happens because in these rooms, the 3 anchor nodes with the strongest signals usually form a linear organization, which makes positioning calculation difficult by least-squares algorithms. On the other hand, in almost every room, our SynTra Avg and SynTra WAvg data fusion solutions had the lowest average error compared to the model-based IPS, and best individual dataset. In this case, the largest average error obtained by SynTra WAvg was 3.16 m in room 2, still resulting in position estimates close to the real position in the same room.

FIGURE 15. Heat-map of the average errors for all test points in the scenario.

In order to better visualize our solution, Figure 15 shows a heat-map of the SynTra WAvg errors in the whole scenario. In this heat-map, we can see problematic regions of the scenario, such as rooms 1, 2, and 11. In these rooms, the worst performance is due to the positions and lack of anchors coverage. In our scenario, the anchor nodes were fixed in locations where it was somewhat convenient to connect them to the power supply. Thus, increasing the density of anchor nodes and centralizing them in the rooms is an alternative to reduce the positioning error.

D. COMPUTATIONAL COSTS ANALYSIS

In this part, we discuss the computational costs of our solution. The most significant and sensitive part is the position estimation that uses a fingerprint dataset. This dataset is composed of s samples measurements for each m RPs and n different anchor nodes. Therefore, the dataset total size is $(s * m * n)$.

During the online phase, signals from a mobile device are used to estimate its position. As mentioned earlier, the algorithm used for this process is KNN. The complexity of KNN depends on the size of the input dataset [22]. Thus, in the traditional fingerprint method, the cost of estimating the mobile device position is $O(s * m * n)$. However, our approach creates T different synthetic datasets. In this case, a mobile device will be classified by KNN into T different datasets. Thus, the complexity of our approach is greater, when compared to the traditional fingerprint, since it involves one more variable T, thus being $O(T * s * m * n)$. There is still the data fusion cost, but it is at most $O(T)$ and, thus, can be ignored.

As we can see, our proposed solution requires a higher processing load compared to traditional fingerprint systems that run KNN only once. This is a key aspect since it limits the number of position estimations per second. However, since it is possible to combine several samples to be classified at the same time using vector implementations of KNN, this limitation can be eased when running in parallel on different CPU cores or even using GPUs on a dedicated IPS server.

VI. DISCUSSION

Fingerprint-based IPSs have an extensive training phase that collects signal strengths at different reference points to create a fingerprint radio map. This technique does not require any prior knowledge of the scenario for radio map creation. However, this radio map needs to be re-created in the presence of changes in the scenario, such as changes in the walls and insertion of new obstacles, making it unfeasible to be maintained for large scenarios. On the other hand, to reduce training cost, our solution requires a small effort to get the floor plan information, and it also needs prior knowledge of the anchor nodes' positions to generate the synthetic training dataset. Despite this effort, the located mobile devices could be displayed on the same map of the area, which could be used as the floor plan.

One can argue that the datasets generated by the RSSI simulator do not represent the real behavior of the signal for the whole scenario. However, when we generate a set of synthetic datasets using different parameters for the signal propagation model, we try to approximate the real signal distribution dynamically in different regions of the system. Some datasets can result in the best estimations in some areas, while other datasets will have better results in other areas. The proposed data fusion techniques try to combine the best estimations into a single solution. Also, in our experiments, we only considered 2D environments. For more complex environments, such as multiple floors, the log-distance model can be easily extended to also include the higher loss from the floors and ceilings.

In our experiments, to perform the comparison with the traditional fingerprint, we used 150 different RPs 2 m apart from each other. It is known that increasing the density of RPs, decreases the average error at the cost of increasing the workload needed for the fingerprint collection. Our solution can create an unrestricted number of virtual RPs, and generate denser datasets, possibly lowering the average error. However, in this case, the performance evaluation could only be done by simulation, which would not be ideal to represent the real scenario. Another important issue is regarding the mobile devices. In our experiments, we used 11 different devices but with the same hardware from the same manufacturer. However, when using mobile devices with different hardware, such as different smartphones, the signal behavior can vary. We believe that our use of several propagation parameters, combined with data fusion, might consider these hardware differences, resulting in better results than traditional fingerprint-based IPSs. In this work, this aspect was not evaluated and will be studied in future works.

VII. CONCLUSION

In fingerprint-based IPSs, building the training dataset in the offline phase is an expensive and complex task. To reduce the effort of data collection, we propose and evaluate a new fingerprint-based IPS, that uses a signal propagation model to generate several synthetic training datasets. We propose four new techniques for the online phase that use data fusion from the position estimations obtained through the different synthetic datasets to estimate a single, more precise position.

Our experiments in a real-world scenario, show two significant contributions: (1) the use of several synthetic datasets to characterize the signal in different regions of the scenario without the need for complex data gathering from the environment, and (2) the use of data fusion techniques to compute the final position of the mobile device. Our performance evaluation shows that SynTra resulted in an average error of 2.36 m, being almost 20% lower than the best, single synthetic dataset, 35% better than a model-based solution, and only 0.48 m from a traditional fingerprint-based IPS, the best-case scenario.

In future works, we intend to experiment with other signal propagation models for the RSSI simulation. We also intend to evaluate the performance of different machine learning algorithms other than KNN. Finally, we intend to propose and evaluate the performance of other data fusion techniques and crowdsourcing.

REFERENCES

[1] M. S. Anwar, F. Hossain, N. Mehajabin, M. Mamun-Or-Rashid, and M. A. Razzaque, "A comparative study on gaussian process regression-based indoor positioning systems," in *2018 International Conference on Innovation in Engineering and Technology (ICIET)*, 2018, pp. 1–5.

[2] S. He and S.-H. G. Chan, "Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 466–490, 2016.

[3] S. Sadowski and P. Spachos, "Rssi-based indoor localization with the internet of things," *IEEE Access*, vol. 6, pp. 30 149–30 161, 2018.

[4] A. Khalajmehrabadi, N. Gatsis, and D. Akopian, "Modern wlan fingerprinting indoor positioning methods and deployment challenges," *IEEE Communications Surveys Tutorials*, vol. 19, no. 3, pp. 1974–2002, 2017.

[5] Z. E. Khatab, A. Hajihoseini, and S. A. Ghorashi, "A fingerprint method for indoor localization using autoencoder based deep extreme learning machine," *IEEE Sensors Letters*, vol. 2, no. 1, pp. 1–4, 2018.

[6] R. Melamed, "Indoor localization: Challenges and opportunities," in *2016 IEEE/ACM International Conference on Mobile Software Engineering and Systems (MOBILESoft)*, 2016, pp. 1–2.

[7] J. Liu, "An ir-uwb indoor positioning based on toa and band-gap modulations," in *2017 International Applied Computational Electromagnetics Society Symposium (ACES)*, 2017, pp. 1–2.

[8] Y. Cheng and T. Zhou, "Uwb indoor positioning algorithm based on tdoa technology," in *2019 10th International Conference on Information Technology in Medicine and Education (ITME)*, 2019, pp. 777–782.

[9] B. Yimwadsana and V. Serey, "Improving accuracy of an aoa-based wi-fi indoor localization using kalman filter," in *2020 17th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 2020, pp. 155–159.

[10] Y. Assayag, H. Oliveira, E. Souto, R. Barreto, and R. Pazzi, "Indoor positioning system using dynamic model estimation," *Sensors*, vol. 20, no. 24, p. 7003, Dec 2020.

[11] G. Li, E. Geng, Z. Ye, Y. Xu, J. Lin, and Y. Pang, "Indoor positioning algorithm based on the improved rssi distance model," *Sensors*, vol. 18, no. 9, p. 2820, Aug 2018.

[12] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '05. New York, NY, USA: Association for Computing Machinery, 2005, p. 205–218.

[13] T. Serif, O. K. Perente, and Y. Dalan, "Robomapper: An automated signal mapping robot for rssi fingerprinting," in *2019 7th International Conference on Future Internet of Things and Cloud (FiCloud)*, 2019, pp. 364–370.

[14] T. Tan, L. Zhang, and Q. Li, "An efficient fingerprint database construction approach based on matrix completion for indoor localization," *IEEE Access*, vol. 8, pp. 130 708–130 718, 2020.

[15] V. Moghtadaiee, S. A. Ghorashi, and M. Ghavami, "New reconstructed database for cost reduction in indoor fingerprinting localization," *IEEE Access*, vol. 7, pp. 104 462–104 477, 2019.

[16] K. T. Herring, J. W. Holloway, D. H. Staelin, and D. W. Bliss, "Path-loss characteristics of urban wireless channels," *IEEE Transactions on Antennas and Propagation*, vol. 58, no. 1, pp. 171–177, Jan 2010.

[17] C. Mohammed El Amine and M. Ouslim, "The implementation of indoor localization based on an experimental study of rssi using a wireless sensor network," *Peer-to-Peer Networking and Applications*, vol. 9, 06 2015.

[18] A. Lazaro, D. Girbau, P. Moravek, and R. Villarino, "Localization in wireless sensor networks using diversity for multipath effects mitigation," in *European Workshop on Smart Objects: Systems, Technologies and Applications*, 2010, pp. 1–6.

[19] X. Fang and L. Chen, "An optimal multi-channel trilateration localization algorithm by radio-multipath multi-objective evolution in rss-ranging-based wireless sensor networks," *Sensors*, vol. 20, no. 6, p. 1798, Mar 2020.

[20] Q. Li, W. Li, W. Sun, J. Li, and Z. Liu, "Fingerprint and assistant nodes based wi-fi localization in complex indoor environment," *IEEE Access*, vol. 4, pp. 2993–3004, 2016.

[21] Y. Shi, W. Shi, X. Liu, and X. Xiao, "An rssi classification and tracing algorithm to improve trilateration-based positioning," *Sensors*, vol. 20, no. 15, p. 4244, Jul 2020.

[22] D. Praveen Kumar, T. Amgoth, and C. S. R. Annavarapu, "Machine learning algorithms for wireless sensor networks: A survey," *Information Fusion*, vol. 49, pp. 1–25, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S156625351830277X

[23] P. Bahl and V. Padmanabhan, "Radar: an in-building rf-based user location and tracking system," in *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*, vol. 2, 2000, pp. 775–784 vol.2.

[24] J. Torres, O. Belmonte, R. Montoliu, S. Trilles, and A. Calia, "How feasible is wifi fingerprint-based indoor positioning for in-home monitoring?" in *2016 12th International Conference on Intelligent Environments (IE)*, 2016, pp. 68–75.

[25] L. Chen, K. Yang, and X. Wang, "Robust cooperative wi-fi fingerprint-based indoor localization," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1406–1417, 2016.

[26] J. Niu, B. Wang, L. Cheng, and J. J. P. C. Rodrigues, "Wicloc: An indoor localization system based on wifi fingerprints and crowdsourcing," in *2015 IEEE International Conference on Communications (ICC)*, 2015, pp. 3008–3013.

[27] P. S. Maher and R. A. Malaney, "A novel fingerprint location method using ray-tracing," in *GLOBECOM 2009 - 2009 IEEE Global Telecommunications Conference*, 2009, pp. 1–5.

[28] R. Kubota, S. Tagashira, Y. Arakawa, T. Kitasuka, and A. Fukuda, "Efficient survey database construction using location fingerprinting interpolation," in *2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA)*, 2013, pp. 469–476.

[29] M. Nowicki and J. Wietrzykowski, "Low-effort place recognition with wifi fingerprints using deep learning," in *Automation 2017*, R. Szewczyk, C. Zieliński, and M. Kaliczyńska, Eds. Cham: Springer International Publishing, 2017, pp. 575–584.

[30] K. S. Kim, S. Lee, and K. Huang, "A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on wi-fi fingerprinting," *Big Data Analytics*, vol. 3, 04 2018.

[31] M. Ali, S. Hur, and Y. Park, "Locali: Calibration-free systematic localization approach for indoor positioning," *Sensors*, vol. 17, p. 1213, 05 2017.

[32] S. Jung, C. oh Lee, and D. Han, "Wi-fi fingerprint-based approaches following log-distance path loss model for indoor positioning," in *2011 IEEE MTT-S International Microwave Workshop Series on Intelligent Radio for Future Personal Terminals*, 2011, pp. 1–2.

[33] C. Qiang, S. Van de Velde, and H. Steendam, "How to get the best out of your fingerprint database: hierarchical fingerprint indoor positioning for databases with variable density," *IEEE Access*, pp. 1–1, 2019.

[34] D. Plets, W. Joseph, K. Vanhecke, E. Tanghe, and L. Martens, "Simple Indoor Path Loss Prediction Algorithm and Validation in Living Lab Setting," *Wireless Personal Communications*, vol. 68, no. 3, pp. 535–552, Feb. 2013.

[35] V. Cantón Paterna, A. Calveras Augé, J. Paradells Aspas, and M. Pérez Bullones, "A bluetooth low energy indoor positioning system with channel diversity, weighted trilateration and kalman filtering," *Sensors*, vol. 17, no. 12, p. 2927, Dec 2017.

[36] K. Huang, K. He, and X. Du, "A hybrid method to improve the ble-based indoor positioning in a dense bluetooth environment," *Sensors*, vol. 19, p. 424, 01 2019.

[37] J. Glitza, P. Zhang, M. Abdelaal, and O. Theel, "An improved ble indoor localization with kalman-based fusion: An experimental study," *Sensors*, vol. 17, p. 951, 04 2017.

[38] M. Bullmann, T. Fetzer, F. Ebner, M. Ebner, F. Deinzer, and M. Grzegorzek, "Comparison of 2.4 ghz wifi ftm- and rssi-based indoor positioning methods in realistic scenarios," *Sensors*, vol. 20, no. 16, p. 4515, Aug 2020.

YURI ASSAYAG received the B.Sc. degree in computer science, and M.Sc. degree in informatics from the Federal University of Amazonas (UFAM), Brazil, in 2020 and 2021, respectively. He is currently pursuing a doctoral degree in informatics at the Federal University of Amazonas. His research interests include wireless communications, indoor positioning systems, and machine learning.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3105188, IEEE Access

HORÁCIO OLIVEIRA received the M.Sc and Ph.D degrees in Computer Science from University of Minas Gerais (UFMG), Brazil, in 2005 and 2008, respectively. He is currently an associate professor at the Federal University of Amazonas, Brazil. His research interest includes vehicular and sensor networks, indoor positioning systems, internet of things, distributed algorithms, and wireless networks.

EDUARDO SOUTO completed his undergraduate studies in Computer Science in 1993 at Federal University of Amazonas (UFAM), Brazil. In 1995, he became a specialist in business administration from the Getúlio Vargas Foundation (FGV). In 2001, he received his M.Sc. degree from the Federal University of Minas Gerais, Brazil, and, in 2007, his Ph.D from the Federal University of Pernambuco, Brazil. Currently, Dr. Eduardo Souto is an Associate Professor at the Institute of Computing of the Federal University of Amazonas. His research interests are in the areas of network security, wireless communication, internet traffic measurement, wireless sensor network, and distributed systems.

RAIMUNDO BARRETO received the M.Sc. and Ph.D degrees in Computer Science from University of Minas Gerais (UFMG) and University of Pernambuco (UFPE), Brazil, in 1997 and 2005, respectively. He is an associate professor at the Federal University of Amazonas' Institute of Computing, where he leads the complex embedded systems research group. His research interest includes embedded and real-time systems, computer architecture, wireless sensor networks, and model-driven engineering.

RICHARD PAZZI received the Ph.D. degree from the University of Ottawa, Canada, in 2008. He is currently an Assistant Professor at the Faculty of Business and Information Technology, University of Ontario Institute of Technology, Canada. Prior to this, he was Post Doc at Ottawa and DIVA Network Manager. His current research interests include communication protocols for vehicular and sensor networks, mobile data gathering in wireless sensor networks, and networked 3D virtual environments.