IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 22, 2018, accepted June 25, 2018, date of publication July 3, 2018, date of current version July 30, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2852658

HybLoc: Hybrid Indoor Wi-Fi Localization Using Soft Clustering-Based Random Decision Forest Ensembles

BEENISH A. AKRAM[1], ALI H. AKBAR[1], AND OMAIR SHAFIQ[2]
[1]Department of Computer Science and Engineering, University of Engineering and Technology, Lahore 54890, Pakistan
[2]Carleton School of Information Technology, Carleton University, Ottawa, ON K1S5B6, Canada

Corresponding author: Beenish A. Akram (beenish.ayesha.akram@uet.edu.pk)

This work was funded by Carleton University and NSERC under Grant Number 201806312.

ABSTRACT Indoor localization has garnered the attention of researchers over the past two decades due to diverse and numerous applications. The existing works either provide room-level or latitude-longitude prediction instead of a hybrid solution, catering only to specific application needs. This paper proposes a new infrastructure-less, indoor localization system named HybLoc using Wi-Fi fingerprints. The system employs Gaussian Mixture Model (GMM)-based soft clustering and Random Decision Forest (RDF) ensembles for hybrid indoor localization i.e., both room-level and latitude-longitude prediction. GMM-based soft clustering allows finding natural data subsets helping cascaded classifiers better learn underlying data dynamics. The RDF ensembles enhance the capabilities of decision trees providing better generalization. A publically available Wi-Fi fingerprints data set UJIIndoorLoc (multi-floor and multi-building) has been used for experimental evaluation. The results describe the potential of HybLoc to provide the hybrid location of user viz a viz the reported literature for both levels of prediction. For room estimation, HybLoc has demonstrated mean 85% accuracy, 89% precision as compared with frequently used k Nearest Neighbors (kNN) and Artificial Neural Network (ANN)-based approaches with 56% accuracy, 60% precision and 42% accuracy, 48% precision, respectively, averaged over all buildings. We also compared HybLoc performance with baseline Random Forest providing 79% accuracy and 82% precision which clearly demonstrates the enhanced performance by HybLoc. In terms of latitude-longitude prediction, HybLoc, kNN, ANN, and baseline Random Forest had 6.29 m, 8.1 m, 180.7 m, and 10.2 m mean error over complete data set. We also present useful results on how number of samples and missing data replacement value affect the performance of the system.

INDEX TERMS Big data applications, indoor localization, machine learning, random decision forest (RDF), ensemble learning, soft clustering.

I. INTRODUCTION

Lots of efforts from academia as well as industry have been put into indoor localization due to the prevalence of smart devices demanding context aware applications. The most important context is the location of a person. Humongous Locations Based Services (LBS) such as healthcare, smart transportation, accident prevention, and evacuation plans in case of terrorist attacks etc., can all benefit from the accurate location provided by an Indoor Positioning System (IPS). In GPS-deprived indoor environments, localization has been extensively explored using various sensory signals such as Wi-Fi [1]–[3], Bluetooth [4], [5], Bluetooth Low Energy (BLE) [6], RFID [7], [8], Ultra wide band signals [9], and images [10] etc. These signals have been employed based on Angle of Arrival (AOA) [11], [12], Time of Arrival (TOA), Time Difference of Arrival (TDOA), Pedestrian Dead Reckoning (PDR) [13], Propagation Model (PM), and fingerprinting approaches. Infrastructure-based and infrastructure-less are the two broad categories in terms of sensory inputs required by these indoor positioning systems. Wi-Fi being infrastructure-less stands out in sensory signals due to pre-existing large scale deployments, almost everywhere, barring the need of additional hardware installations. Fingerprinting based solutions are favored because techniques such as TOA and AOA require specialized antennae along with strict time synchronization [14]. PDR suffers from error propagation

in successive location estimates. Furthermore, propagation model based methods majorly rely on the estimated distances from a Wi-Fi Access Point (AP) to a user for location estimation using trilateration, degrading its performance in real world scenarios. Hence in this paper, we propose an IPS using fingerprints (FPs) of Wi-Fi signals.

A. CONTRIBUTIONS

1) Most of the existing works on indoor localization report their results either for room-level prediction [7], [15], [16] or in terms of latitude-longitude [17], [18] or any other explicit coordinates. These two approaches cannot be compared directly because even a prediction error of one meter in terms of x, y coordinates can localize the person either in the actual room or the one adjacent to it. This misjudgment has non-trivial implications for applications with specific requirements such as precise room-level accuracy. Consequently, the literature on indoor localization is broadly categorized into two namely, room–level prediction and latitude-longitude prediction (translated into meters). We present a new IPS based on soft clustering and ensemble of ensembles which provides location in terms of both latitude-longitude and room-level prediction, integrating major parallel streams of indoor localization.
2) Partitioning of dataset in existing work has either been done based on clustering Reference Points (RPs) into disjoint groups rather than clustering dataset samples [7], [19]–[22] or hard clustering of dataset samples [23], [24]. Dataset samples partitioning into overlapping and/or non-overlapping subsets has been performed based on mere AP visibility in a sample reading [15], [25] which results in as many data subsets as there are number of APs in the dataset and the same number of trained classifiers. In such a mechanism, the number of classifiers for all clusters will linearly increase with growing number of APs visible in a building resulting in many classifiers' invocations per prediction. We propose a new dataset samples partitioning approach where GMM based soft clustering is employed, guided by Akaike Information Criteria (AIC) and Bayesian Information Criteria (BIC) to find natural groups in dataset samples. This approach also allows the system designers to control the number of trained classifiers as well as maximum number of classifiers invoked per prediction.
3) The existing works on IPS mostly use their own proprietary datasets which are far smaller in size, number of users responsible for data collection, device diversity, and are usually not publically accessible. Effectively such works are rendered unusable in order to reproduce and/or compare results with other works. We present results on *UJIIndoorLoc*, a publically available dataset containing 21,048 Wi-Fi FPs of 520 APs marked with ground truth, collected by 20 users, and 25 different android devices ensuring validation of real world scenarios. It contains FPs of 3 buildings of University at Jaume I, Spain. Each building has 4 or more floors, covering an area of almost 110,000m^2. Our reported results can be validated with various other existing indoor localization systems that have utilized the same dataset.
4) We report the impact of missing RSSI value replacement in Wi-Fi samples and report interesting findings on it.
5) We also report the effect of number of samples per room utilized in training. Results are reported on unfiltered and filtered datasets based on our proposal of room-sample frequency based thresholding mechanism.

B. OUTLINE

The organization of the paper is as follows: Section II discusses related work focusing mainly on Wi-Fi based localization approaches and IPS utilizing *UJIIndoorLoc* dataset. In Section III, a summarized overview of the dataset is provided to let the reader know the experimental area details, predictors, and ground truth labels of the samples. The proposed localization system is described in Section IV, both in terms of training and location prediction phases. In Section V, the results of the proposed localization system (HybLoc) on the dataset are provided in comparison with most widely used kNN, ANN and baseline Random Forest based approaches for indoor localization. Finally, conclusion is presented in Section VI.

II. RELATED WORK

Numerous IPS have been presented in recent years. Concerning our work, two aspects of indoor positioning are relevant, indoor localization using Wi-Fi fingerprints and indoor localization using *UJIIndoorLoc* dataset.

A. WORK BASED ON WI-FI LOCALIZATION

RADAR [26] from Microsoft® research lab is the pioneer work using Wi-Fi signals and radio propagation model on indoor location estimation. It utilized Wi-Fi FPs collected at the Wi-Fi Access Points (APs) of the laptop carried by a user. Triangulation and k-nearest neighbors both were utilized to approximate x, y coordinates of user, reporting 2-3m median error. The experimental area was just a single floor of 43.5m $\times$ 22.5m (980m^2) dimensions with only 3 APs resulting in 0.003 APs/m^2. They reported results primarily in the form of Cumulative Distribution Function (CDF) of the positioning error along with 25th, 50th, 75th, and 95th percentile error in meters.

Kanaris *et al.* [27] utilized hybrid sensory input consisting of visible light and radio signals. They proposed filtering of dataset based on visible light communication (VLC) and then modified kNN was used on that data subset for final location estimation. Their results discussed performance both with Wi-Fi only and Wi-Fi with VLC indicating mean error reduction from 4.7m to 1.89m when VLC is used with 20% of the

total dataset size for computing prediction. An area of 160m^2 was covered by 6 APs (0.03 APs/m^2). Their approach is not completely infrastructure-less as they identify the region of interest in the first step using VLC which requires specialized hardware. They presented results on merely 7 specific test points. An average positioning error in meters was presented on each such test point. They compared their proposed method against kNN based approach using Wi-Fi only.

Sun *et al.* [10] combined Wi-Fi signals with camera images to optimize propagation model parameters, using trilateration and Wi-Fi fingerprints. The crowdsourced Wi-Fi fingerprints were utilized to adjust for localization errors in trilateration. Furthermore, panoramic camera and room map were used to detect human object on the observed image to find its pixel location. The pixel location was then mapped to the room map using ANN. Their results were in the form of x, y coordinates with mean error of 3.15m in a corridor and a single room. Their approach for crowdsourcing the data required 2-D code stickers for identification of place with the submitted user FPs. It also required installation of panoramic cameras for location prediction. Their experimental area of 51.6m $\times$ 20.4m (1052.64m^2) had 7 APs translating into 0.007 APs/m^2. They expressed their results in a specific room and specific room + corridor in terms of mean positioning error in meters and cumulative probability within both 1m and 2m. They mainly compared their results with kNN method for indoor localization.

Cooper *et al.* [28] made use of FPs using Wi-Fi combined with Bluetooth Low Energy (BLE) radio signals. Modified AdaBoost algorithm in conjunction with Decision Stumps was applied for room-level location estimation. They trained a classifier per room in One-vs-All notion. They presented results with both Wi-Fi only and Wi-Fi + BLE in their approach called Loco. They reported 94% accuracy using Wi-Fi only. When Wi-Fi + BLE combined signals were used, it increased to 96%. However, AdaBoost is a boosting technique that cannot be parallelized for training as well as predicting. The One-vs-All notion computation required for every room also makes Loco's response time dependent on the building size and the number of rooms per building. The response time of Loco worsens directly in proportion to the number of rooms. Their experimental setup covered an area of 1,900m^2 with dense coverage of 159 APs, resulting in 0.08 APs/m^2. They compared their results with Redpin [29] in terms of accuracy for room level prediction, utilizing a combination of GSM cell information, Bluetooth, and Wi-Fi signals.

Li *et al.* [20], proposed affinity propagation clustering combined with Particle Swarm Optimization based ANN for each cluster. Data dimension reduction was performed using Principle Component Analysis (PCA) before clustering. They presented results in terms of x, y coordinates. They reported mean error of 1.89m and 90% error of 2.9m on experimental area of 45m $\times$ 25m (1,125m^2) with 16 APs (0.014 APs/m^2).

Song *et al.* [21], focused on elimination of redundant APs for each reference point (RP), based on best discriminating APs selection. They employed modified ReliefF with Pearson's correlation coefficient for APs elimination followed by clustering on the filtered data. RP clustering was based on threshold of minimum size of common subset of best discriminating APs. Then a Hidden Naïve Bayes (HNB) model was trained for each cluster. To estimate location, cluster matching and respective HNB was invoked to estimate x, y coordinates. Mean error of 1.68m with 2.21 standard deviation in positioning error was reported by the authors. The experimental area of 800m^2 was covered by 50 APs (0.06 APs/m^2).

Górak *et al.* [15] focused on two things; one, finding important APs using Random Forest. Second, proposing a scheme to determine malfunctioning APs during operation. They evaluated their proposed system in normal and malfunctioning APs scenarios. For floor detection, they reported an error rate of 4% and 2 meters for horizontal detection, instead of 30% and 7m without malfunctioning APs detection mechanism. An area of 50m $\times$ 70m (3,500m^2) was covered for experiments with a total of 570 APs (0.16 APs/m^2). Górak *et al.* [25] proposed an IPS employing Random Forest with a new take on dataset partitioning in the same experimental set up. They generated subsets according to RSSI signal visibility of each AP. All observations in their dataset with non-missing values of an AP's RSSI were included in that AP's subset, resulting in number of subsets equal to number of APs. A Random Forest was trained for each subset for x, y coordinates prediction and floor prediction, reporting mean error of 3.1m and 0.04 (absolute floor number difference) respectively. They compared their proposed approach with baseline Random Forest approach and with multilayer perceptron indicating 5-9% improvement in mean horizontal error, whereas floor detection accuracy remained the same.

Belmonte-Fernández *et al.* [16], proposed an IPS based on Wi-Fi fingerprinting for room-level localization, targeting ambient assisted living (AAL) as an application area. Their experiments focused on evaluating performance based on combination of training and testing data under posture variations (standing/sitting), making a total of 4 combinations, and utilized numerous classifiers and their proposed ensemble classifier to present their results. They evaluated their proposed system in 5 different apartments of various sizes specifically 120m^2 with 33 visible APs (0.27 APs/m^2), 80m^2 with 36 visible APs (0.45 APs/m^2), 90m^2 with 27 visible APs (0.3 APs/m^2), 80m^2 with 43 visible APs (0.53 APs/m^2), and 62m^2 with 23 visible APs (0.37 APs/m^2). They used accuracy as the only performance measure. They showed that different classifiers were suitable for different combinations. They reported however that the maximum accuracy of 76.7%, averaged over all 5 scenarios, was achieved only by Random Forest.

B. WORK BASED ON UJIINDOORLOC

The dataset covered an area of 110,000m^2 with total 520 Wi-Fi APs visible during data collection from all

buildings (0.004 APs/m^2). Wietrzykowski *et al.* [23] used visual space identification algorithm FAB-MAP for indoor localization using Wi-Fi FPs. They presented results in the form of x, y coordinates. They reported accuracy as a measure of correct prediction of both Building ID and floor ID combined i.e., both were identified correctly. They reported error in meters between actual and predicted location with minimum 8.21m for only those samples for which both building ID and floor ID were predicted correctly. However, such performance measure evaluation leaves out results on those samples' positioning error for which either building ID or floor ID was incorrectly predicted. Furthermore, no comparison with any other existing approach was reported.

Torres-Sospedra *et al.* [18] reported results on x, y coordinates prediction along with floor and building prediction. They provided two different datasets of magnetic field (*UJIIndoorLoc-Mag*) and Wi-Fi RSSI covering the same area. Basic kNN was used for both magnetic and Wi-Fi RSSI values. Mean positioning error for magnetic field based discrete and continuous methods in the reported 11 testing paths was 7.23m and 6.05m respectively. For Wi-Fi dataset, mean error of 4.54m was presented with minimum error of 4.27m. They reported results in terms of mean positioning error in meters and response time in seconds. Their main focus was on presenting a new dataset as the primary contribution, therefore comparisons with existing approaches were not drawn on their provided dataset.

Bozkurt *et al.* [30] used the dataset to investigate different classifiers for various levels of predictions i.e. building, floor and region level which is their definition of a new attribute composed of a triplet consisting of Building ID, Floor ID, and Space ID. For building level prediction, they compared BayesNet, Sequential Minimal Optimization (SMO), Artificial Neural Network (ANN), J48, and Naïve Bayes with BayesNet providing best accuracy of 99.8%. For predicting floor and region level, ANN was the winner with 89.9% accuracy. They used accuracy and response time as the performance evaluation measures.

Uddin and Islam [31] proposed the usage of extremely randomized trees for x, y coordinates prediction. Their reported Root Mean Squared Error (RMSE) of the proposed approach was 12.21m for longitude and 10.12m for latitude. For building and floor level prediction 100% and 91.44% accuracy was attained. They evaluated building ID and floor ID prediction using accuracy/success rate, and for latitude-longitude prediction they used RMSE and normalized RMSE. Nowicki and Wietrzykowski [32] used the RSSI values to hierarchically perform building and floor identification using deep learning. They reported an accuracy of 91% for correct identification of building and floor classification.

An overview of the existing work highlights the need for a unified approach which caters for the needs of applications requiring meter level location identification as well as room-level prediction.

III. BRIEF DESCRIPTION OF *UJIINDOORLOC* DATASET

The dataset was presented by Torres-Sospedra *et al.* [33]. The dataset was collected at three buildings of University Jaume I, Madrid, Spain. Each building contained four or more floors and total covered area was 110,000m^2. A total of 529 attributes in the provided 21,048 Wi-Fi FPs consist of 520 Wi-Fi AP RSSI values, Building ID, Floor ID, Space ID, latitude, longitude, user ID who collected the data, device ID describing the phone's manufacturer along with model, and date/time stamp. As it contains building, floor, and space IDs along with latitude and longitude, it can be used for both classification (building/floor/space prediction) and/or regression (determining latitude-longitude values). Twenty different users, using 25 different Android devices, created this dataset. The dataset consists of 19,937 training samples and 1,111 test samples.

The RSSI values of the APs varied from -104dBm (weak signal $\sim$ far AP) to 0dBm (strong signal $\sim$ near AP). As all APs are not visible at all locations, resulting dataset is sparse with numerous missing values. These missing values are labeled with value $+100$ in the original dataset.

The rationale for using this dataset is twofold: first, it readily allows the reader to directly compare the results with existing IPS using the same dataset instead of results on a small, proprietary dataset. Second, most of the reported works collect a dataset from a rather small area (usually a research lab floor/ portion of departmental building) which does not depict a real world scenario. This dataset is large enough to let the IPS show its capability in true sense. Consider there are a total of M APs detected in the complete dataset. The dataset consists of total R rows of the following format termed as *FPi*. Each row in the dataset is a fingerprint *FPi* where $FPi = \{x_1, x_2, x_3, \ldots, x_M, \}$ and x_j represents the received signal strength from jthAP in the collected sample. As ground truth, 3 labels are tagged with each such sample namely Room_i, Lat_i and $Long_i$ representing Room ID, latitude and longitude values respectively.

IV. HybLoc

Our proposed system targets indoor localization at building level since either GPS or AP MAC address matching can be easily used to narrow down to building level. The main idea here is to split the dataset for a building using soft clustering performed by Gaussian Mixture Model (GMM) into overlapping and/or non-overlapping data subsets comprising of similar observations. These subsets are then assigned to different subsystems specifically customized to process the respective data employing Random Decision Forest (RDF) ensembles [34]. Many recent research contributions indicate that combining clustering and classification ensembles can yield a better and improved classifier as clustering can impose useful constraints on the classification task [35]–[38]. This was the motivation behind combining clustering and classifier ensembles, where clustering is applied first to FP samples to group similar observations together. Then classification ensembles are grown for room-level prediction whereas

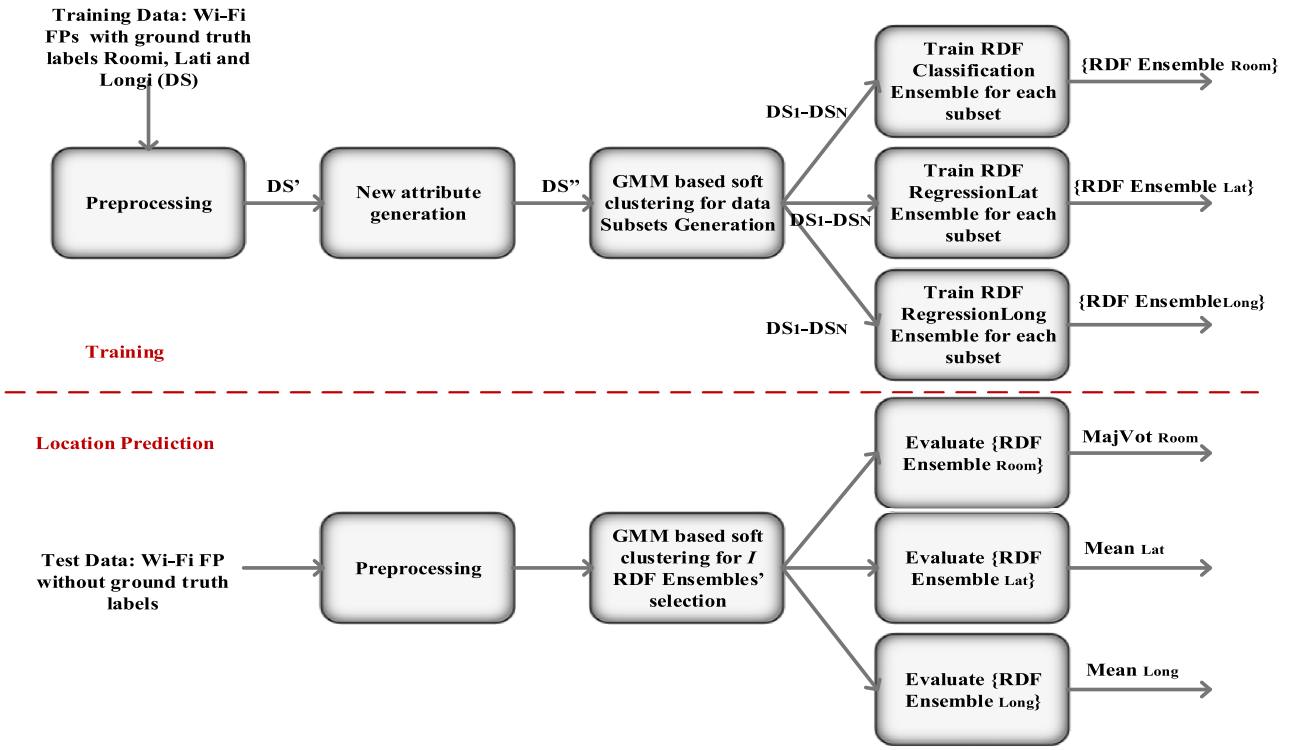

FIGURE 1. Proposed IPS (HybLoc).

regression ensembles are used for latitude-longitude prediction. Merely 3 hyper parameters comprising number of trees to be grown (TreeNum), random number of predictors as the basis of split (f), and the maximum number of splits (depth of trees: SplitsMax) are needed to be tuned for training a RDF ensemble. Therefore, Random Forest is suitable for rapid and repeated training, required for practical and real world deployment of localization systems. RDF ensemble was selected because it is suitable both in terms of accuracy and efficiency on large datasets, robust with respect to noise, can handle missing values and generalizes well too. It uses bootstrapping which results in reduced variance without raising the bias because different partitions of training dataset with replacement ensure that the decision trees are uncorrelated. Being an ensemble learning method, it combines the strengths of weak learners (Decision Trees) to enhance its generalization capability. Moreover, its training and prediction both can be parallelized for reduced time consumption. The fluctuation of Wi-Fi fingerprints at the same RP due to persons/things crossing by, weather conditions, even the occlusion caused by person holding a Wi-Fi enabled device [9], [39] etc., does not make it suitable for RP clustering. Moreover, the clustering of data samples/Wi-Fi FPs is a better choice as it helps distinguish different groups of FPs. One classifier trained per cluster is better able to learn the data subset dynamics rather than one classifier learning the whole dataset. Instead of providing any fix notion and mechanism (number of clusters fixed e.g. equal to number of APs), our approach allows dictation of both inherent data dynamics as well as administrator control over finding suitable number of clusters within the dataset. GMM considers the variances within the cluster itself and allows soft clustering based on probability of a sample belonging to more than one cluster. The reason behind employing GMM based soft clustering of dataset samples instead of hard clustering of RPs or samples is that GMM distribution and Wi-Fi propagation characteristics are very close in nature except for the peak extremely near to AP location [40], hence GMM is a very good candidate for Wi-Fi RSSI samples clustering. The experimental results also validate our approach. The holistic working of the system is presented in Fig. 1. Fig. 2, Fig. 3, and Fig. 4 describe the training and prediction phases. The details of each phase are presented as follows:

A. TRAINING

Training phase is also called off-line phase in which the system is prepared using the training dataset. The following steps summarized in Fig. 2 were carried out during training:

1) Data Preprocessing
2) New Attribute Generation
3) *N* Data Subsets Generation
4) *N* RDF Ensemble Classification Training for Room Prediction
5) *N* RDF Ensemble Regression Training for Latitude-Longitude Prediction

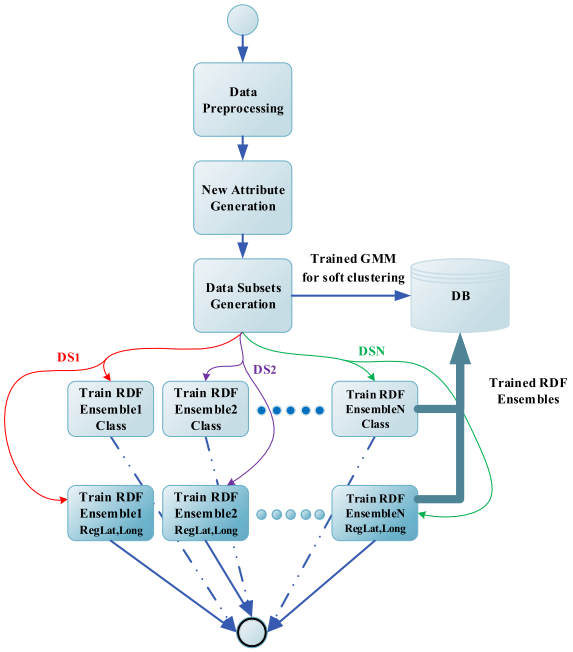

FIGURE 2. Training phase.

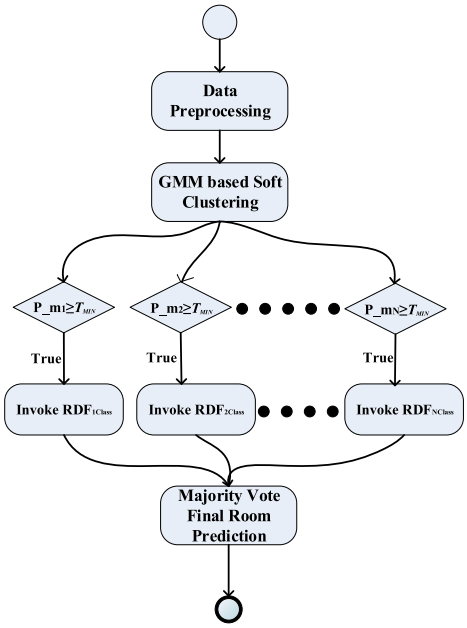

FIGURE 3. Location prediction phase for room prediction.

1) DATA PREPROCESSING

Data preprocessing usually includes filling in the missing values and alteration of data representation. In the dataset, the missing values of AP RSSI are represented with value +100dBm. In majority of existing FP based IPS, the missing values are replaced with a value slightly smaller than the weakest RSSI value in the dataset. We used missing values +100dBm of the dataset. Moreover, we varied the missing value from −105dBm to −110dBm (best performance obtained at −110dBm) whose results are presented in Section V.

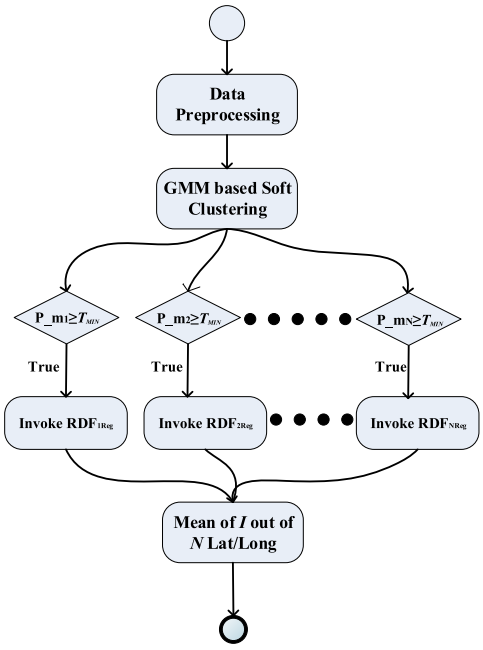

FIGURE 4. Location prediction phase for coordinates prediction, one set of ensembles depicted here for each latitude and longitude estimation.

2) NEW ATTRIBUTE GENERATION

We were interested in coordinates prediction as well as room-label prediction. The data labeling for room label prediction had three relevant fields namely Building ID (3 buildings), Floor ID (4 floors in building 0, 1, and 5 floors in building 2), and Space ID. These Floor IDs and Space IDs were redundant in these buildings so the triplet of all three fields was required to identify a particular room. We generated a new attribute named Room ID used for room label training and prediction, instead of this triplet combination, to uniquely identify a particular room out of a total of 735 rooms in all 3 buildings.

3) *N* DATA SUBSETS GENERATION

Data subsets were obtained by applying soft clustering to each building's dataset samples using GMM. GMM assigns label as well as cluster membership probabilities (*P_m*) to each sample. Based on these probabilities soft clustering of data is possible by threshold application. Several parameters of GMM were adjusted to find suitable *N* soft clusters which includes number of clusters (numeric value), covariance type (diagonal or full), and covariance sharing (true or false). The parameter tuning was performed in light of both Akaike Information Criteria (AIC) and Bayesian Information Criteria (BIC) being minimum along with the final performance evaluation parameters obtained. AIC and BIC were computed based on optimized log likelihood value (*L*), number of parameters (*numParam*), and number of observations (*numObs*) in the dataset using (1) and (2).

$$AIC = -2(\mathrm{LOG}(L)) + 2(numParam) \quad (1)$$

$$BIC = -2(\mathrm{LOG}(L)) + numParam * \mathrm{LOG}\ numObs \quad (2)$$

TABLE 1. Algorithm I: pseudocode of proposed algorithm for training.

Input:
DS = {FP_1, FP_2, …… FP_R}
where $FPi = \{x_1, x_2, x_3, ..., x_{M,}\}$, $x_j = RSSI\ AP_j$,
CovType= covariance type (diagonal or full),
CovSharing = covariance sharing (true or false),
M_{Val}= missing value replacement,
N_{Max}= maximum number of clusters,
T_{MIN} = Cluster membership minimum threshold
Output:
N data subsets,
Trained GMM,
N trained RDF $Ensembles_{class}$,
N trained RDF $Ensembles_{RegLat}$,
N trained RDF $Ensembles_{RegLong}$

1: Replace empty RSSI values with M_{Val}
2: Identify Unique BuildingID, FloorID, SpaceID combinations
3: Generate New Attribute RoomID from step 2, label the dataset
4: for i : 1 -> N_{Max}
5: for each *CovSharing* ϵ true, false
6: for each *CovType* ϵ Diagonal, Full
7: Invoke GMM for clusters formation
8: for o: 1 -> *R*
9: if $P\_m_o >= T_{MIN}$
10: Include Sample in Corresponding Cluster data subset
11: end if
12: end for
13: Compute AIC, BIC using (1) and (2)
14: Save Data Subsets, GMM Model
15: end for
16: end for
17: end for
18: *C* Configs :=Based on Minimum AIC & BIC, shortlist configs
19: for each c ϵ *C* Configs
20: for each Dataset
21: Train Classification RDF ensemble (Algorithm III):
22: Find optimal $Tree_{Num,}$ $Splits_{Max,}$ *f*
23: Compute Room Level Performance Measures
24: Save *N* trained RDF ensembles for classification
25: Train Regression RDF ensemble for Lat., Long.(Algorithm III):
26: Find optimal $Tree_{Num,}$ $Splits_{Max,}$ *f*
27: Compute Euclidean Positioning Error Performance Measures
28: Save *N* trained RDF ensembles for regression for Latitude
29: Save *N* trained RDF ensembles for regression for Longitude
30: end for
31: end for
32: Select and save best *N* classification ensembles $RDFEnsemble_{Class}$
33: Select and save best *N* regression ensembles $RDFEnsemble_{RegLat}$
34: Select and save best *N* regression ensembles $RDFEnsemble_{RegLong}$
35: Save corresponding pre-trained soft GMM

TABLE 2. Algorithm II: pseudocode of proposed algorithm for location prediction.

Input:
$FPi = \{x_1, x_2, x_3, ..., x_{M,}\}$, $xj = RSSI\ AP_j$,
M_{Val}= missing value replacement,
T_{MIN} = Cluster membership minimum threshold
Output:
$Room_i$,
Lat_i,
$Long_i$

1: Replace empty RSSI values with M_{Val}
2: Invoke pre-trained GMM for cluster membership probabilities
3: For Room Prediction:
4: Load All *N* pre-trained RDF ensembles for classification
5: for each n: 1 -> *N*
6: If $P\_m_{Sample} >= T_{MIN}$
7: Invoke nth $RDFEnsemble_{class}$ to predict $Room_n$ (Algorithm III)
8: end if
9: end for
10: $Room_i$ = Majority Vote {$Room_n$}
11: For Latitude Prediction:
12: Load All *N* pre-trained RDF ensembles for latitude
13: for each n: 1 -> *N*
14: If $P\_m_{Sample} >= T_{MIN}$
15: Invoke nth $RDFEnsemble_{RegLat}$ to predict Lat_n (Algorithm III)
16: end if
17: end for
18: Lat_i = Mean {Lat_n}
19: For Longitude Prediction:
20: Load All *N* pre-trained RDF ensembles for longitude
21: for each n: 1 -> *N*
22: If $P\_m_{Sample} >= T_{MIN}$
23: Invoke nth $RDFEnsemble_{RegLong}$ to predict $Long_n$ (Algorithm III)
24: end if
25: end for
26: $Long_i$ = Mean {$Long_n$}

The initial centroids of clusters were determined by using k-means++ algorithm. Afterwards, the samples' membership to different clusters/subsets was determined on the basis of minimum threshold (T_{min}) compared with P_m. The trained GMM and resulting data subsets were saved for further use at Location Prediction stage. All the ground truth fields were kept intact during this partitioning procedure including Building ID, Floor ID, Space ID, latitude-longitude values, and the new attribute, named Room ID. Concerning Building 0, optimal performance of RDF ensembles for classification and regression was obtained at 2 clusters, full covariance, and shared covariance set as true with minimum threshold 0.4. For Building 1 and 2, it was 2 clusters, diagonal covariance, shared covariance set as true with minimum threshold 0.4 and 2 clusters, full covariance, shared covariance as false with 0.3 minimum threshold.

4) *N* RDF ENSEMBLE CLASSIFICATION TRAINING FOR ROOM PREDICTION

For each building, the generated data subsets from step 3 were used to train RDF ensembles for room-level prediction in ratio of 70-30% stratified training and testing datasets. For each RDF ensemble 300 trees, 25 random features, and 1,024 maximum splits per tree were found to be providing optimal results. The training was performed using 10-fold cross validation on 70% training subset with Room ID as the ground truth label. It must be noted that for each building there were *N* data subsets and corresponding *N* RDF ensembles trained per subset which were saved to be used in prediction stage.

5) *N* RDF ENSEMBLE REGRESSION TRAINING FOR LATITUDE-LONGITUDE PREDICTION

For each building, the very same data subsets were used to train *N* RDF ensembles on 70% stratified training portion but for latitude-longitude prediction based on regression instead of classification. Separate RDF ensemble with 300 trees, 25 random features, and 1,024 maximum splits per tree,

TABLE 3. Algorithm III: pseudocode of RDF ensemble for training and location prediction (room, coordinate level).

Input:
Training data subset with total M predictors,
Number of Trees $Tree_{Num}$,
Maximum Number of Splits $Splits_{Max}$,
Random Number of Predictors f
Output:
Predicted Room location L_x/ Predicted Coordinate (Lat/Long) $L_{Lat/Long}$
For Training:
1: for $i = 1$ to $Tree_{Num}$
2: From the training dataset, select a bootstrap sample set S of size TD with replacement
3: Produce a Random Forest Tree T_i to S, by recursively iterating the points 4-6 for each terminal node of the tree, until the maximum number of splits $Splits_{Max}$ is reached
4: Randomly pick f predictors from the M predictors ($f << M$)
5: Select the best predictor/split-point among the f
6: Split the node into two child nodes
7: end for
8: Output the ensemble of trees $\{T_i\}_1^{TreeNum}$
For Room prediction at a new point x from RDF $L_{cl}^{TreeNum}$:
9: Assume $L_j(x)$ be the room prediction of the j^{th} Random Forest tree
10: $L_x = L_{cl}^{TreeNum}(x)$ = majority vote $\{L_j(x)\}_1^{TreeNum}$
For Latitude/Longitude prediction at a new point x from RDF $L_{reg}^{TreeNum}$:
11: $L_{Lat/Long} = L_{reg}^{TreeNum}(x) = \frac{1}{TreeNum}\sum_{i=1}^{TreeNum} T_i(x)$

was trained for latitude and for longitude ground truth label, later on the latitude-longitude results were combined using Euclidean distance formula given in (3) for positioning error (*PosError*) calculation in meters where *pr* and *gt* imply predicted and ground truth values respectively.

$$PosError = \sqrt{(Lat_{pr} - Lat_{gt})^2 + (Long_{pr} - Long_{gt})^2} \quad (3)$$

B. LOCATION PREDICTION

Location prediction phase is the online phase in which the FP sample from a user is captured and processed to estimate the unknown location. It is pictorially represented in Fig. 3 and Fig. 4, and consists of the following four steps:

1) Data Preprocessing
2) Soft Cluster Membership determination
3) Invocation of associated I RDF Ensemble for Room Prediction
4) Invocation of associated I RDF Ensemble for Latitude-Longitude Prediction

1) DATA PREPROCESSING

During location prediction, the missing values in the collected Wi-Fi RSSI sample were replaced with the missing value chosen in training phase. If missing values +100dBm were used during training, then +100 will be placed in location prediction phase too.

2) SOFT CLUSTER MEMBERSHIP DETERMINATION

The stored pre-trained GMM from training phase, step 3 was used to determine the membership probabilities (P_m) of the sample at hand.

TABLE 4. Building 0 room level prediction results.

IPS	Dataset	Accuracy	Precision	Recall
HybLoc	UnfltrdMV100	0.74	0.74	0.74
kNN	UnfltrdMV100	0.40	0.49	0.40
Base-RF	UnfltrdMV100	0.72	0.71	0.71
ANN, 2-L, SCG	UnfltrdMV100	0.46	0.52	0.46
ANN, 2-L, RBP	UnfltrdMV100	0.32	0.55	0.32
ANN, 3-L, SCG	UnfltrdMV100	0.43	0.56	0.42
ANN, 3-L, RBP	UnfltrdMV100	0.30	0.50	0.30
ANN, 4-L, SCG	UnfltrdMV100	0.40	0.48	0.39
ANN, 4-L, RBP	UnfltrdMV100	0.27	0.40	0.26
HybLoc	UnfltrdMVn110	0.82	0.85	0.82
kNN	UnfltrdMVn110	0.46	0.57	0.46
Base-RF	UnfltrdMVn110	0.79	0.82	0.79
ANN, 2-L, SCG	UnfltrdMVn110	0.47	0.54	0.47
ANN, 2-L, RBP	UnfltrdMVn110	0.50	0.51	0.49
ANN, 3-L, SCG	UnfltrdMVn110	0.38	0.60	0.37
ANN, 3-L, RBP	UnfltrdMVn110	0.31	0.35	0.31
ANN, 4-L, SCG	UnfltrdMVn110	0.33	0.34	0.32
ANN, 4-L, RBP	UnfltrdMVn110	0.27	0.30	0.27
HybLoc	FltrdMV100	0.75	0.79	0.75
kNN	FltrdMV100	0.40	0.50	0.40
Base-RF	FltrdMV100	0.72	0.75	0.72
ANN, 2-L, SCG	FltrdMV100	0.47	0.51	0.47
ANN, 2-L, RBP	FltrdMV100	0.32	0.37	0.32
ANN, 3-L, SCG	FltrdMV100	0.45	0.47	0.44
ANN, 3-L, RBP	FltrdMV100	0.45	0.51	0.45
ANN, 4-L, SCG	FltrdMV100	0.40	0.45	0.40
ANN, 4-L, RBP	FltrdMV100	0.25	0.31	0.24
HybLoc	FltrdMVn110	0.83	0.85	0.82
kNN	FltrdMVn110	0.47	0.56	0.47
Base-RF	FltrdMVn110	0.79	0.82	0.79
ANN, 2-L, SCG	FltrdMVn110	0.54	0.57	0.53
ANN, 2-L, RBP	FltrdMVn110	0.34	0.40	0.33
ANN, 3-L, SCG	FltrdMVn110	0.42	0.62	0.42
ANN, 3-L, RBP	FltrdMVn110	0.31	0.44	0.31
ANN, 4-L, SCG	FltrdMVn110	0.38	0.40	0.38
ANN, 4-L, RBP	FltrdMVn110	0.28	0.35	0.27

3) INVOCATION OF ASSOCIATED I RDF ENSEMBLES FOR ROOM PREDICTION

The same minimum threshold (T_{min}) value applied in training phase was used to determine the membership to different clusters/subsets. The membership (P_m) to different N clusters was further used to invoke I (clusters whose membership satisfies the condition: $P\_m >= T_{min}$) out of N pretrained RDF classification ensembles for room estimation. The final room/class label was based on majority vote from all invoked ensembles. In case of a tie in majority voting, the final decision was made by selecting the prediction produced by clusters/subsets with higher cluster membership probability (P_m) obtained in the step 2 of location prediction phase.

4) INVOCATION OF ASSOCIATED I RDF ENSEMBLE FOR LATITUDE-LONGITUDE PREDICTION

Following the same pattern used for room prediction, minimum threshold value (T_{min}) applied on soft cluster membership (P_m) was used to select relevant regression RDF ensembles. Separate set of RDF ensembles was invoked for latitude and longitude value estimation (I out of N regression ensembles for latitude prediction and I out of N regression ensembles for longitude prediction). The final prediction of

TABLE 5. Building 0 room level training and response time.

IPS	Dataset	Training Time(s)	Response Time(s)
HybLoc	UnfltrdMV100	68.64	1.16E-01
kNN	UnfltrdMV100	-	7.30E-04
Base-RF	UnfltrdMV100	51.97	5.73E-03
ANN, 2-L, SCG	UnfltrdMV100	85.55	2.57E-05
ANN, 2-L, RBP	UnfltrdMV100	79.29	2.48E-05
ANN, 3-L, SCG	UnfltrdMV100	115.93	3.20E-05
ANN, 3-L, RBP	UnfltrdMV100	57.22	3.06E-05
ANN, 4-L, SCG	UnfltrdMV100	110.78	3.75E-05
ANN, 4-L, RBP	UnfltrdMV100	85.83	3.94E-05
HybLoc	UnfltrdMVn110	69.30	9.03E-02
kNN	UnfltrdMVn110	-	7.40E-04
Base-RF	UnfltrdMVn110	49.77	5.82E-03
ANN, 2-L, SCG	UnfltrdMVn110	83.69	2.41E-05
ANN, 2-L, RBP	UnfltrdMVn110	105.27	2.40E-05
ANN, 3-L, SCG	UnfltrdMVn110	94.74	2.96E-05
ANN, 3-L, RBP	UnfltrdMVn110	71.65	2.95E-05
ANN, 4-L, SCG	UnfltrdMVn110	109.87	3.68E-05
ANN, 4-L, RBP	UnfltrdMVn110	67.40	3.59E-05
HybLoc	FltrdMV100	62.16	3.41E-02
kNN	FltrdMV100	-	7.20E-04
Base-RF	FltrdMV100	46.98	5.37E-03
ANN, 2-L, SCG	FltrdMV100	65.72	2.38E-05
ANN, 2-L, RBP	FltrdMV100	73.79	2.52E-05
ANN, 3-L, SCG	FltrdMV100	76.23	2.92E-05
ANN, 3-L, RBP	FltrdMV100	57.00	2.78E-05
ANN, 4-L, SCG	FltrdMV100	90.06	3.51E-05
ANN, 4-L, RBP	FltrdMV100	54.41	3.82E-05
HybLoc	FltrdMVn110	56.26	7.36E-03
kNN	FltrdMVn110	-	7.25E-04
Base-RF	FltrdMVn110	45.38	5.33E-03
ANN, 2-L, SCG	FltrdMVn110	65.96	2.42E-05
ANN, 2-L, RBP	FltrdMVn110	73.36	2.21E-05
ANN, 3-L, SCG	FltrdMVn110	76.47	2.82E-05
ANN, 3-L, RBP	FltrdMVn110	40.64	2.92E-05
ANN, 4-L, SCG	FltrdMVn110	91.36	3.51E-05
ANN, 4-L, RBP	FltrdMVn110	77.30	4.66E-05

TABLE 6. Building 1 room level prediction results.

IPS	Dataset	Accuracy	Precision	Recall
HybLoc	UnfltrdMV100	0.84	0.83	0.80
kNN	UnfltrdMV100	0.58	0.53	0.55
Base-RF	UnfltrdMV100	0.82	0.77	0.77
ANN, 2-L, SCG	UnfltrdMV100	0.64	0.71	0.60
ANN, 2-L, RBP	UnfltrdMV100	0.43	0.47	0.40
ANN, 3-L, SCG	UnfltrdMV100	0.63	0.68	0.57
ANN, 3-L, RBP	UnfltrdMV100	0.57	0.62	0.51
ANN, 4-L, SCG	UnfltrdMV100	0.61	0.64	0.57
ANN, 4-L, RBP	UnfltrdMV100	0.20	0.26	0.17
HybLoc	UnfltrdMVn110	0.86	0.84	0.81
kNN	UnfltrdMVn110	0.66	0.60	0.62
Base-RF	UnfltrdMVn110	0.83	0.78	0.78
ANN, 2-L, SCG	UnfltrdMVn110	0.70	0.74	0.65
ANN, 2-L, RBP	UnfltrdMVn110	0.42	0.47	0.37
ANN, 3-L, SCG	UnfltrdMVn110	0.63	0.68	0.55
ANN, 3-L, RBP	UnfltrdMVn110	0.39	0.41	0.33
ANN, 4-L, SCG	UnfltrdMVn110	0.61	0.65	0.53
ANN, 4-L, RBP	UnfltrdMVn110	0.19	0.25	0.15
HybLoc	FltrdMV100	0.85	0.85	0.82
kNN	FltrdMV100	0.58	0.64	0.57
Base-RF	FltrdMV100	0.81	0.83	0.78
ANN, 2-L, SCG	FltrdMV100	0.67	0.71	0.64
ANN, 2-L, RBP	FltrdMV100	0.62	0.66	0.59
ANN, 3-L, SCG	FltrdMV100	0.65	0.63	0.61
ANN, 3-L, RBP	FltrdMV100	0.64	0.61	0.60
ANN, 4-L, SCG	FltrdMV100	0.64	0.67	0.61
ANN, 4-L, RBP	FltrdMV100	0.37	0.39	0.33
HybLoc	FltrdMVn110	0.87	0.89	0.85
kNN	FltrdMVn110	0.67	0.70	0.65
Base-RF	FltrdMVn110	0.85	0.82	0.82
ANN, 2-L, SCG	FltrdMVn110	0.75	0.77	0.71
ANN, 2-L, RBP	FltrdMVn110	0.20	0.24	0.17
ANN, 3-L, SCG	FltrdMVn110	0.69	0.73	0.63
ANN, 3-L, RBP	FltrdMVn110	0.40	0.45	0.35
ANN, 4-L, SCG	FltrdMVn110	0.65	0.69	0.59
ANN, 4-L, RBP	FltrdMVn110	0.38	0.41	0.34

latitude and longitude was generated by taking mean of all latitude values and mean of all longitude values respectively.

Training and location prediction phases of the proposed method (HybLoc) are formally described in form of Algorithm I and Algorithm II in Table. 1 and Table. 2 respectively.

Training and location prediction phases of Random Decision Forest ensemble are formally described in form of Algorithm III in Table. 3

Equation (4) describes a 2-dimensional Gaussian distribution where μ is the mean and Σ is the covariance matrix. A Gaussian Mixture Model having N number of overlapping Gaussian distributions is represented by (5) and (6).

$$N(x \mid \mu, \Sigma) = \frac{1}{2\pi\sqrt{|\Sigma|}}\exp\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\} \tag{4}$$

$$P(x) = \sum_{k=1}^{N} \pi_k N\left(x \mid \mu_k, \Sigma_k\right) \tag{5}$$

$$\sum_{k=1}^{N} \pi_k = 1 \tag{6}$$

Mixing coefficient is represented by π_k and expresses each mixing element's weight. Where the summation of all the mixing coefficients is equal to 1. The contour of the 2-D Gaussian distribution is determined by the individual Gaussian distribution average, covariance and mixing matrices. Provided, the linearly-mixed weighted coefficients of each distribution average and covariance are tuned employing a sufficient number of Gaussian distributions, any arbitrary, continuous density function may be approximated.

C. HybLoc TIME COMPLEXITY OF TRAINING AND PREDICTION

Training and prediction time complexity of HybLoc can be derived in the following manner.

1) TIME COMPLEXITY OF TRAINING

The time complexity of training an unpruned Decision Tree (DT) is expressed in (7).

$$O(M \times Rlog(R)) \tag{7}$$

Where

M = number of predictors,

R = number of observations/samples

TABLE 7. Building 1 room level training and response time.

IPS	Dataset	Training Time(s)	Response Time(s)
HybLoc	UnfltrdMV100	45.95	4.66E-03
kNN	UnfltrdMV100	-	6.92E-04
Base-RF	UnfltrdMV100	36.89	4.20E-03
ANN, 2-L, SCG	UnfltrdMV100	52.89	2.45E-05
ANN, 2-L, RBP	UnfltrdMV100	23.25	2.4E-05
ANN, 3-L, SCG	UnfltrdMV100	54.17	2.69E-05
ANN, 3-L, RBP	UnfltrdMV100	35.15	2.63E-05
ANN, 4-L, SCG	UnfltrdMV100	68.22	3.42E-05
ANN, 4-L, RBP	UnfltrdMV100	41.85	3.21E-05
HybLoc	UnfltrdMVn110	46.30	2.96E-02
kNN	UnfltrdMVn110	-	7.00E-04
Base-RF	UnfltrdMVn110	38.55	4.27E-03
ANN, 2-L, SCG	UnfltrdMVn110	43.14	1.97E-05
ANN, 2-L, RBP	UnfltrdMVn110	47.02	1.99E-05
ANN, 3-L, SCG	UnfltrdMVn110	54.19	2.58E-05
ANN, 3-L, RBP	UnfltrdMVn110	33.76	2.72E-05
ANN, 4-L, SCG	UnfltrdMVn110	67.47	3.36E-05
ANN, 4-L, RBP	UnfltrdMVn110	32.41	3.43E-05
HybLoc	FltrdMV100	42.60	5.35E-01
kNN	FltrdMV100	-	6.86E-04
Base-RF	FltrdMV100	32.94	3.65E-03
ANN, 2-L, SCG	FltrdMV100	30.69	2.01E-05
ANN, 2-L, RBP	FltrdMV100	21.07	1.92E-05
ANN, 3-L, SCG	FltrdMV100	41.47	2.4E-05
ANN, 3-L, RBP	FltrdMV100	23.30	2.83E-05
ANN, 4-L, SCG	FltrdMV100	54.37	3.11E-05
ANN, 4-L, RBP	FltrdMV100	32.71	3.33E-05
HybLoc	FltrdMVn110	42.75	2.15E+00
kNN	FltrdMVn110	-	6.73E-04
Base-RF	FltrdMVn110	32.58	3.61E-03
ANN, 2-L, SCG	FltrdMVn110	30.84	1.97E-05
ANN, 2-L, RBP	FltrdMVn110	23.18	2.02E-05
ANN, 3-L, SCG	FltrdMVn110	41.39	2.61E-05
ANN, 3-L, RBP	FltrdMVn110	26.79	2.5E-05
ANN, 4-L, SCG	FltrdMVn110	54.64	3.13E-05
ANN, 4-L, RBP	FltrdMVn110	31.15	3.08E-05

TABLE 8. Building 2 room level prediction results.

IPS	Dataset	Accuracy	Precision	Recall
HybLoc	UnfltrdMV100	0.79	0.83	0.77
kNN	UnfltrdMV100	0.47	0.48	0.45
Base-RF	UnfltrdMV100	0.75	0.90	0.73
ANN, 2-L, SCG	UnfltrdMV100	0.52	0.54	0.49
ANN, 2-L, RBP	UnfltrdMV100	0.35	0.38	0.33
ANN, 3-L, SCG	UnfltrdMV100	0.43	0.49	0.39
ANN, 3-L, RBP	UnfltrdMV100	0.33	0.39	0.30
ANN, 4-L, SCG	UnfltrdMV100	0.40	0.46	0.36
ANN, 4-L, RBP	UnfltrdMV100	0.29	0.35	0.26
HybLoc	UnfltrdMVn110	0.84	0.86	0.82
kNN	UnfltrdMVn110	0.55	0.56	0.52
Base-RF	UnfltrdMVn110	0.82	0.83	0.80
ANN, 2-L, SCG	UnfltrdMVn110	0.39	0.50	0.33
ANN, 2-L, RBP	UnfltrdMVn110	0.37	0.44	0.31
ANN, 3-L, SCG	UnfltrdMVn110	0.31	0.38	0.25
ANN, 3-L, RBP	UnfltrdMVn110	0.27	0.31	0.21
ANN, 4-L, SCG	UnfltrdMVn110	0.27	0.31	0.21
ANN, 4-L, RBP	UnfltrdMVn110	0.22	0.28	0.16
HybLoc	FltrdMV100	0.79	0.83	0.76
kNN	FltrdMV100	0.47	0.51	0.45
Base-RF	FltrdMV100	0.79	0.88	0.76
ANN, 2-L, SCG	FltrdMV100	0.53	0.57	0.50
ANN, 2-L, RBP	FltrdMV100	0.56	0.59	0.53
ANN, 3-L, SCG	FltrdMV100	0.47	0.53	0.43
ANN, 3-L, RBP	FltrdMV100	0.51	0.54	0.47
ANN, 4-L, SCG	FltrdMV100	0.41	0.46	0.37
ANN, 4-L, RBP	FltrdMV100	0.27	0.33	0.23
HybLoc	FltrdMVn110	0.84	0.92	0.81
kNN	FltrdMVn110	0.55	0.55	0.52
Base-RF	FltrdMVn110	0.80	0.88	0.77
ANN, 2-L, SCG	FltrdMVn110	0.44	0.47	0.38
ANN, 2-L, RBP	FltrdMVn110	0.29	0.32	0.24
ANN, 3-L, SCG	FltrdMVn110	0.35	0.38	0.29
ANN, 3-L, RBP	FltrdMVn110	0.32	0.38	0.26
ANN, 4-L, SCG	FltrdMVn110	0.30	0.37	0.25
ANN, 4-L, RBP	FltrdMVn110	0.24	0.28	0.20

As RDF ensemble is comprised of many DTs and it uses only a small number f out of total number of predictors M. One DT complexity in RDF is represented by (8) and the complexity of $Tree_{Num}$ by (9)

$$\boldsymbol{O(f \times Rlog(R))} \tag{8}$$

$$\boldsymbol{O(Tree_{Num} \times f \times Rlog(R))} \tag{9}$$

where

$Tree_{Num}$ = number of trees in RDF ensemble,

f = random features selected for tree best split

We are also controlling the depth of the trees grown using $Split_{max}$. Hence training complexity of one RDF ensemble becomes (10).

$$\boldsymbol{O(Tree_{Num} \times f \times R \times Split_{max})} \tag{10}$$

N such RDF ensembles are grown for room prediction, latitude prediction and longitude prediction. Hence for each such N RDF ensembles, the training time complexity is represented by (11).

$$\boldsymbol{O(Tree_{Num} \times f \times R \times Split_{max} \times N)} \tag{11}$$

The training time complexity of GMM is expressed by (12).

$$\boldsymbol{O(R \times K \times D^3)} \tag{12}$$

Where

R = number of observations/samples,

K = number of components,

D = number of dimensions

Hence as per our proposed algorithm, the training time complexity of HybLoc is governed by (13).

$$\boldsymbol{O(R \times K \times D^3) + O(Tree_{Num} \times f \times R \times Split_{max} \times N \times m)} \tag{13}$$

Where m = number of cascaded blocks of ensembles, which is 3 in our case, one for room classification and two for latitude, longitude regression.

2) TIME COMPLEXITY OF PREDICTION

The time complexity of one DT and one RDF ensemble for prediction are shown by (14) and (15) respectively.

$$\boldsymbol{O(Rlog(R))} \tag{14}$$

$$\boldsymbol{O(Tree_{Num} \times Rlog(R))} \tag{15}$$

If $Split_{max}$ is used to control depth of trees, then time complexity of prediction by one RDF ensemble is represented

TABLE 9. Building 2 room level training and response time.

IPS	Dataset	Training Time(s)	Response Time(s)
HybLoc	UnfltrdMV100	123.23	9.42E-03
kNN	UnfltrdMV100	-	8.51E-04
Base-RF	UnfltrdMV100	148.28	3.45E-02
ANN, 2-L, SCG	UnfltrdMV100	204.72	2.59E-05
ANN, 2-L, RBP	UnfltrdMV100	171.08	2.79E-05
ANN, 3-L, SCG	UnfltrdMV100	222.49	3.09E-05
ANN, 3-L, RBP	UnfltrdMV100	138.50	3E-05
ANN, 4-L, SCG	UnfltrdMV100	248.65	3.72E-05
ANN, 4-L, RBP	UnfltrdMV100	200.99	3.86E-05
HybLoc	UnfltrdMVn110	112.30	7.05E-03
kNN	UnfltrdMVn110	-	8.59E-04
Base-RF	UnfltrdMVn110	115.04	1.43E-02
ANN, 2-L, SCG	UnfltrdMVn110	204.27	2.56E-05
ANN, 2-L, RBP	UnfltrdMVn110	233.09	2.56E-05
ANN, 3-L, SCG	UnfltrdMVn110	221.61	3.03E-05
ANN, 3-L, RBP	UnfltrdMVn110	182.25	3.01E-05
ANN, 4-L, SCG	UnfltrdMVn110	247.46	3.91E-05
ANN, 4-L, RBP	UnfltrdMVn110	161.45	3.71E-05
HybLoc	FltrdMV100	108.17	6.72E-03
kNN	FltrdMV100	-	8.17E-04
Base-RF	FltrdMV100	92.71	1.02E-02
ANN, 2-L, SCG	FltrdMV100	151.49	2.62E-05
ANN, 2-L, RBP	FltrdMV100	93.59	2.65E-05
ANN, 3-L, SCG	FltrdMV100	169.15	3.15E-05
ANN, 3-L, RBP	FltrdMV100	99.39	2.8E-05
ANN, 4-L, SCG	FltrdMV100	194.50	3.77E-05
ANN, 4-L, RBP	FltrdMV100	144.07	3.69E-05
HybLoc	FltrdMVn110	103.85	6.46E-03
kNN	FltrdMVn110	-	8.25E-04
Base-RF	FltrdMVn110	95.20	1.17E-02
ANN, 2-L, SCG	FltrdMVn110	151.30	2.32E-05
ANN, 2-L, RBP	FltrdMVn110	199.99	2.67E-05
ANN, 3-L, SCG	FltrdMVn110	169.06	2.82E-05
ANN, 3-L, RBP	FltrdMVn110	119.21	2.91E-05
ANN, 4-L, SCG	FltrdMVn110	194.11	3.72E-05
ANN, 4-L, RBP	FltrdMVn110	123.34	3.59E-05

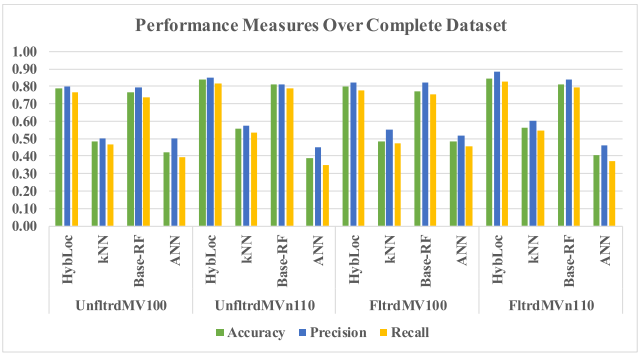

FIGURE 5. Performance measures, accuracy, precision and recall averaged over all 3 buildings in the dataset.

by (16) as follows.

$$O(Tree_{Num} \times R \times Split_{max}) \tag{16}$$

I out of N such RDF ensembles are invoked for each room prediction, latitude prediction and longitude prediction. All N such ensembles can be triggered at maximum. Hence for each such N RDF ensembles, the prediction time complexity

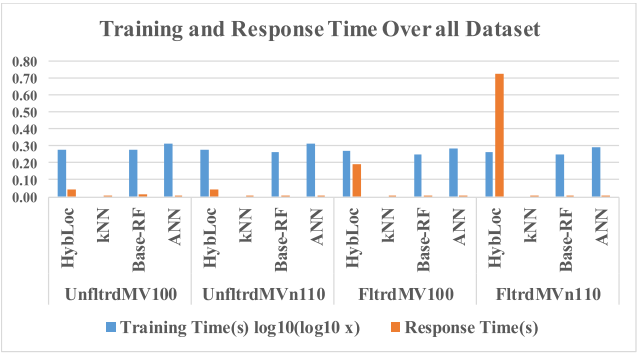

FIGURE 6. Performance measures, training time and response time averaged over all 3 buildings in the dataset.

can be expressed by (17).

$$O(Tree_{Num} \times R \times Split_{max} \times N) \tag{17}$$

The prediction time complexity of GMM is expressed by (18).

$$O(K \times D^3) \tag{18}$$

Hence as per our proposed algorithm, the prediction time complexity of HybLoc is governed by (19).

$$O(K \times D^3) + O(Tree_{Num} \times R \times Split_{max} \times N \times m) \tag{19}$$

V. EXPERIMENTAL EVALUATION

This section describes the experiments conducted to evaluate the performance of HybLoc in terms of both room-level and latitude-longitude prediction. The results for room-level estimation are presented in terms of accuracy, precision, recall, time required for training and time required for testing. Majorly latitude-longitude related results are reported in literature using mean positioning error [41] or Cumulative Distribution Function (CDF) [42]. Positioning error is expressed in the form of estimated Euclidean distance compared with ground truth Euclidean distance. We present minimum Euclidean distance, maximum, mean, mode, standard deviation as well as CDF of the positioning errors obtained over the datasets for latitude-longitude prediction. The results are presented based on building level as previously discussed in Section IV, GPS can be used to narrow down the search to building level easily. The complete dataset includes data for 3 buildings. We first separated the dataset building-wise. Then each building's dataset was partitioned into 70-30% stratified sections used for 10-fold validation during training and 30% unseen data was reserved for separate testing purposes. It was observed during detailed inspection of the dataset that some rooms had very few samples recorded. We filtered the data based on minimum samples per room kept at 19 (rooms with less than 19 samples were discarded from the dataset termed as 'filtered data') to investigate the impact of such low samples in these rooms. Also the default value existing in dataset for missing RSSI values

TABLE 10. Building 0 latitude-longitude level positioning error in meter.

IPS	Dataset	Min	Max	Mean	Mode	Std
HybLoc	UnfltrdMV100	0.13	39.83	6.72	0.13	4.82
kNN	UnfltrdMV100	0.29	42.60	10.17	8.37	7.02
Base-RF	UnfltrdMV100	3.07	37.67	9.65	3.07	4.47
ANN, 2-L, SCG	UnfltrdMV100	0.22	108.80	16.52	0.22	12.82
ANN, 2-L, RBP	UnfltrdMV100	16.09	758.97	299.84	16.09	133.59
ANN, 3-L, SCG	UnfltrdMV100	0.16	57.84	11.42	0.16	8.02
ANN, 3-L, RBP	UnfltrdMV100	76.13	625.92	349.26	76.13	81.18
ANN, 4-L, SCG	UnfltrdMV100	0.21	55.14	11.38	0.21	7.24
ANN, 4-L, RBP	UnfltrdMV100	102.10	464.17	306.64	102.10	59.74
HybLoc	UnfltrdMVn110	0.03	30.12	5.42	0.03	3.80
kNN	UnfltrdMVn110	0.24	41.50	6.29	13.03	4.40
Base-RF	UnfltrdMVn110	3.07	32.37	8.32	3.07	3.74
ANN, 2-L, SCG	UnfltrdMVn110	0.48	112.14	16.03	0.48	12.17
ANN, 2-L, RBP	UnfltrdMVn110	268.12	974.38	574.96	268.12	101.91
ANN, 3-L, SCG	UnfltrdMVn110	0.33	87.80	16.54	0.33	11.40
ANN, 3-L, RBP	UnfltrdMVn110	196.23	700.58	404.39	196.23	74.01
ANN, 4-L, SCG	UnfltrdMVn110	0.25	64.17	12.00	0.25	7.73
ANN, 4-L, RBP	UnfltrdMVn110	129.27	494.46	303.94	129.27	50.37
HybLoc	FltrdMV100	0.15	48.67	6.95	0.15	4.66
kNN	FltrdMV100	0.29	42.30	10.16	7.94	7.00
Base-RF	FltrdMV100	3.09	36.90	9.68	3.09	4.40
ANN, 2-L, SCG	FltrdMV100	0.05	102.28	17.10	0.05	12.82
ANN, 2-L, RBP	FltrdMV100	7.67	507.31	183.31	7.67	79.91
ANN, 3-L, SCG	FltrdMV100	0.29	63.53	13.60	0.29	9.29
ANN, 3-L, RBP	FltrdMV100	48.76	121.97	77.53	57.40	16.32
ANN, 4-L, SCG	FltrdMV100	0.40	87.27	16.40	0.40	11.59
ANN, 4-L, RBP	FltrdMV100	7.19	482.86	195.18	7.19	78.28
HybLoc	FltrdMVn110	0.09	28.76	5.13	0.09	3.40
kNN	FltrdMVn110	0.24	43.73	6.28	0.93	4.52
Base-RF	FltrdMVn110	3.03	31.79	8.33	3.03	3.65
ANN, 2-L, SCG	FltrdMVn110	0.34	106.33	12.23	0.34	9.41
ANN, 2-L, RBP	FltrdMVn110	1.14	399.62	90.58	1.14	64.51
ANN, 3-L, SCG	FltrdMVn110	0.27	79.05	15.08	0.27	10.02
ANN, 3-L, RBP	FltrdMVn110	210.74	724.96	466.61	210.74	81.26
ANN, 4-L, SCG	FltrdMVn110	0.18	64.45	11.70	0.18	7.89
ANN, 4-L, RBP	FltrdMVn110	4.30	459.11	189.57	4.30	77.31

was +100dBm, we found during experiments that with our proposed approach the performance improved with missing values replaced with negative value smaller than the smallest value, best found to be at −110dBm. The rationale behind this

TABLE 11. Building 1 latitude-longitude level positioning error in meter.

IPS	Dataset	Min	Max	Mean	Mode	Std
HybLoc	UnfltrdMV100	0.16	65.15	8.58	0.76	6.24
kNN	UnfltrdMV100	0.36	85.16	12.22	2.81	9.36
Base-RF	UnfltrdMV100	3.26	74.64	11.35	3.28	6.15
ANN, 2-L, SCG	UnfltrdMV100	0.32	256.62	32.08	5.63	28.31
ANN, 2-L, RBP	UnfltrdMV100	12.39	876.68	218.62	190.35	127.72
ANN, 3-L, SCG	UnfltrdMV100	1.01	148.49	20.99	8.81	16.54
ANN, 3-L, RBP	UnfltrdMV100	3.39	596.14	167.00	152.68	100.51
ANN, 4-L, SCG	UnfltrdMV100	0.47	123.37	18.83	5.30	13.62
ANN, 4-L, RBP	UnfltrdMV100	239.07	922.69	515.76	364.99	120.60
HybLoc	UnfltrdMVn110	0.14	73.10	7.82	1.58	6.00
kNN	UnfltrdMVn110	0.45	78.32	10.22	14.51	8.51
Base-RF	UnfltrdMVn110	3.13	71.87	10.65	3.35	5.94
ANN, 2-L, SCG	UnfltrdMVn110	0.31	115.19	17.97	11.48	13.15
ANN, 2-L, RBP	UnfltrdMVn110	4.53	648.24	203.08	223.09	102.08
ANN, 3-L, SCG	UnfltrdMVn110	0.52	107.65	16.98	8.07	11.46
ANN, 3-L, RBP	UnfltrdMVn110	212.90	605.38	307.04	309.43	63.80
ANN, 4-L, SCG	UnfltrdMVn110	0.56	118.35	16.69	9.48	11.72
ANN, 4-L, RBP	UnfltrdMVn110	425.25	886.00	668.78	699.21	67.97
HybLoc	FltrdMV100	0.29	71.19	8.37	0.67	6.18
kNN	FltrdMV100	0.49	84.13	11.99	2.74	9.33
Base-RF	FltrdMV100	3.38	73.34	11.24	3.58	5.98
ANN, 2-L, SCG	FltrdMV100	0.66	216.84	31.97	10.61	26.52
ANN, 2-L, RBP	FltrdMV100	12.25	1141.05	515.07	462.52	208.90
ANN, 3-L, SCG	FltrdMV100	0.43	129.99	24.34	12.75	17.73
ANN, 3-L, RBP	FltrdMV100	111.50	1070.50	626.47	604.73	145.10
ANN, 4-L, SCG	FltrdMV100	0.34	109.17	14.94	2.84	11.71
ANN, 4-L, RBP	FltrdMV100	6.87	728.11	220.82	213.26	118.57

TABLE 11. *(Continued.)* **Building 1 latitude-longitude level positioning error in meter.**

HybLoc	FltrdMVn110	0.00	69.59	7.67	0.47	5.86
kNN	FltrdMVn110	0.45	78.12	10.14	12.59	8.39
Base-RF	FltrdMVn110	3.19	70.03	10.57	3.86	5.78
ANN, 2-L, SCG	FltrdMVn110	0.35	142.22	20.33	11.52	16.19
ANN, 2-L, RBP	FltrdMVn110	253.39	1116.38	667.50	515.41	148.34
ANN, 3-L, SCG	FltrdMVn110	0.43	117.93	20.97	7.66	14.54
ANN, 3-L, RBP	FltrdMVn110	207.81	905.36	469.76	541.12	94.16
ANN, 4-L, SCG	FltrdMVn110	0.24	92.93	15.67	3.96	10.86
ANN, 4-L, RBP	FltrdMVn110	20.27	684.95	357.52	473.20	109.06

approach is simple and logical. The smaller the RSSI value the weaker the signal, hence replacing missing values with +100dBm meant the signal was strongest whereas there was absolutely no signal captured, which caused confusion for the classifier. Hence, the results are presented in 4 folds:

1. First, the results are presented on complete buildings' data with existing missing value in the dataset +100 dBm: UnfltrdMV100
2. Second, the results are presented on complete buildings' data with missing value kept as −110dBm: UnfltrdMVn110
3. Third, the results are presented on filtered data of buildings with missing value +100 dBm: FltrdMV100
4. Fourth, the results are presented on filtered data of buildings with missing value −110dBm: FltrdMVn110

First the results are presented for room-level prediction for each building separately, followed by the averaged overall performance. Then the latitude-longitude prediction results are expressed in the same manner. The results obtained by HybLoc are compared with k Nearest Neighbors (kNN) and Artificial Neural Network (ANN), the most frequently used approaches for indoor localization. Also the performance of HybLoc is compared with Random Forest (same values of parameters) directly applied without GMM clustering on building level dataset referred as Base Random Forest (Base-RF) for fair comparison of advantage that HybLoc presents over straight forward application of Random Forest.

A. ROOM LEVEL PREDICTION RESULTS

The room level results are expressed for each building individually by HybLoc, kNN, Base-RF and ANN. Moreover, mean performance evaluation measures for all buildings are presented. The results expressed for kNN were obtained by taking mean of performance measures by 6 different

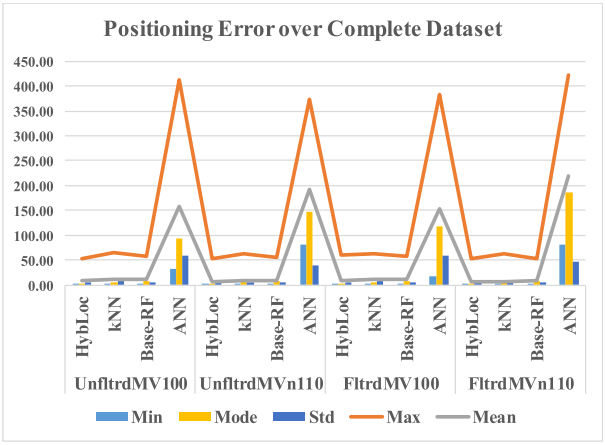

FIGURE 7. Positioning error in meters averaged over all 3 buildings in the dataset.

configurations of kNN related to number of k and distance measure used. The results for ANN were computed for 2-Layer, 3-Layer and 4-Layer networks utilizing Scaled Conjugate Gradient (SCG) and Resilient Back Propagation (RBP) training algorithms averaged over 3 different configurations for each combination i.e. the results presented for 2-Layer network with SCG training algorithm are the mean of 3 different configurations having various number of neurons per hidden layer specifically (100, 200, 500 averaged for 2-Layer, 50-50, 100-100, 500-500 for 3-Layer, and 50-50-50, 100-100-100, 500-500-500 for 4-Layer). Results on Building 0 are presented in Table. 4. It must be noted that all these results are based on system's performance on 30% stratified unseen data kept for testing. The training time includes both GMM clustering time plus time consumed by N RDF ensembles training. Whereas, response time is the summation of GMM clustering time and I pre-trained RDF ensembles' time consumed for each sample on average.

It is evident from Table. 4 that HybLoc performs well in comparison with kNN based approach for room-level prediction. The maximum accuracy achieved for building 0 was 83%. The sheer impact caused by missing value replacement is also evident from it, as on UnfltrdMV100 the accuracy was 73% which rose to 82% with missing value −110 dBm used in the same dataset. Also it can be seen that having more samples for each location (room) helps the system learn better as comparing performance of HybLoc and kNN both performed better in FltrdMV100 than UnfltrdMV100 where accuracy increased from 0.73 to 0.75 for HybLoc but remained same for kNN. The reason for this can be related to only a few discarded rooms in the filtered dataset (26 out of 256 room were filtered based on threshold). All networks of ANN also followed the similar trend performing better with MVn110 than with MV100 and with filtered dataset than unfiltered one. SCG training algorithm was found to be more suitable than RBP with very little outlier cases. However, the overall accuracy obtained by ANN was far lower than HybLoc. HybLoc also clearly wins over Base-RF validating the effectiveness of our proposed approach in all four scenarios.

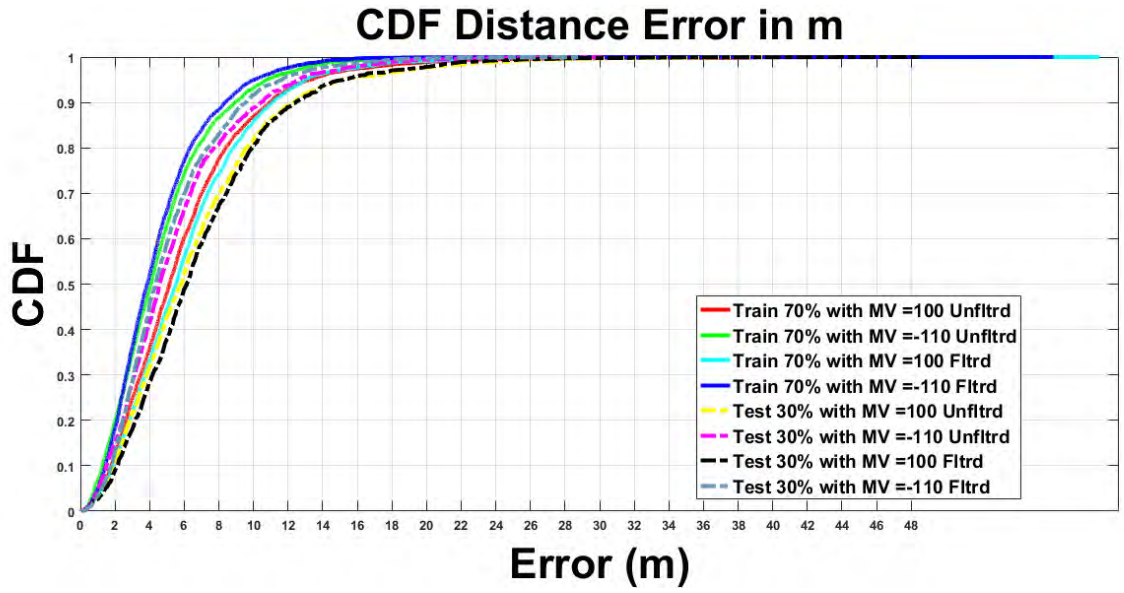

FIGURE 8. CDF of HybLoc for Building 0 10 fold-cross validated 70% training performance along with results on 30% test data.

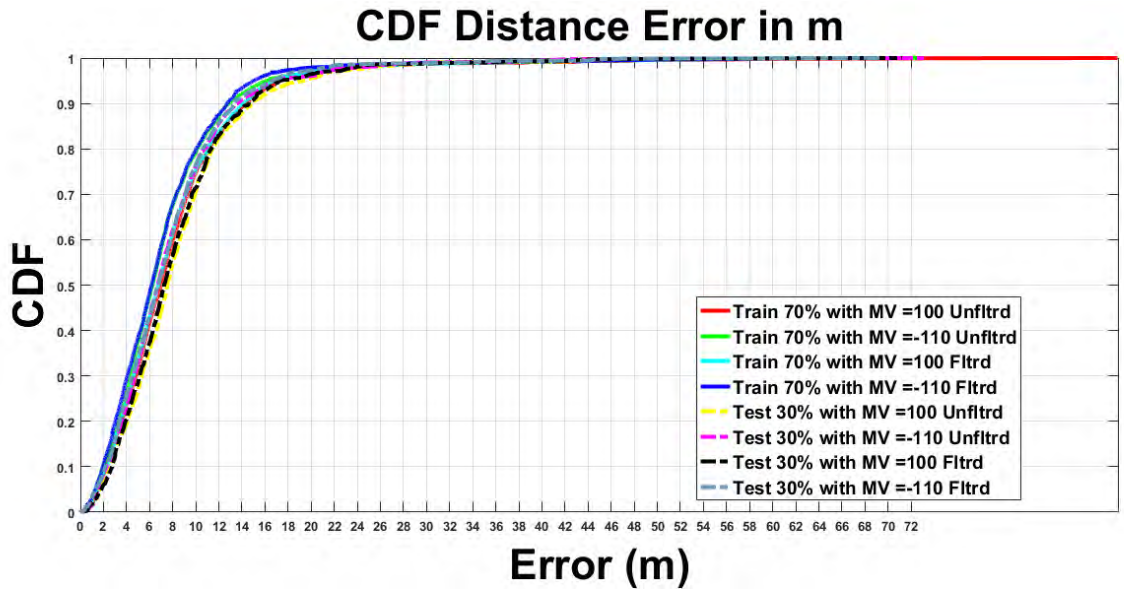

FIGURE 9. CDF of HybLoc for building 1 cross validated 70% training performance along with results on 30% test data.

Results on Building 0 related to training and testing time are presented in Table. 5 in seconds.

Table. 5 sheds light on training time and response time for all compared approaches. kNN does not need any training as being an instance based machine learning approach, it stores all the samples and for prediction searches the whole dataset and k nearest neighbors are included in the majority vote for the final prediction. It is interesting to note that response time of kNN almost remained same for all 4 cases. For HybLoc, it was not the case. For filtered vs unfiltered dataset, it consumed lesser time in training for filtered dataset obviously due to comparatively smaller number of samples. Even more interesting is the impact of filtering data as well as missing value impact. In both cases, the response time was reduced by 10 times with -110 dBm instead of 100 dBm and with filtered dataset instead of unfiltered one. ANN showed minimum response time of the scale of E-05 seconds which remained consistent for all 4 scenarios. It should be noted that training time varied highly for different configurations of ANN. Sometimes SCG consumed more time for training than RBF and vice versa. Training time is also not directly related to number of neurons or number of layers as a

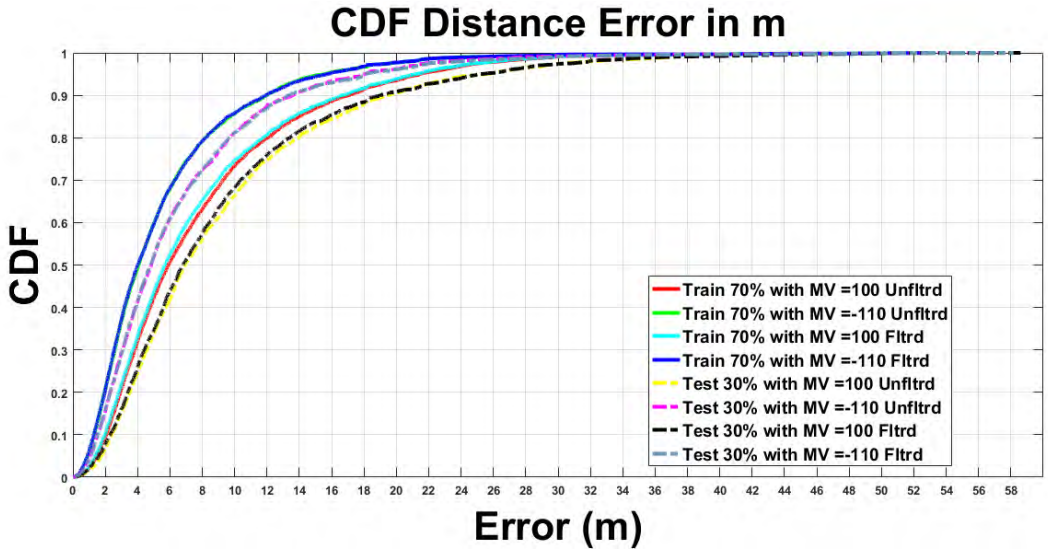

FIGURE 10. CDF of HybLoc for building 2 cross validated 70% training performance along with results on 30% test data.

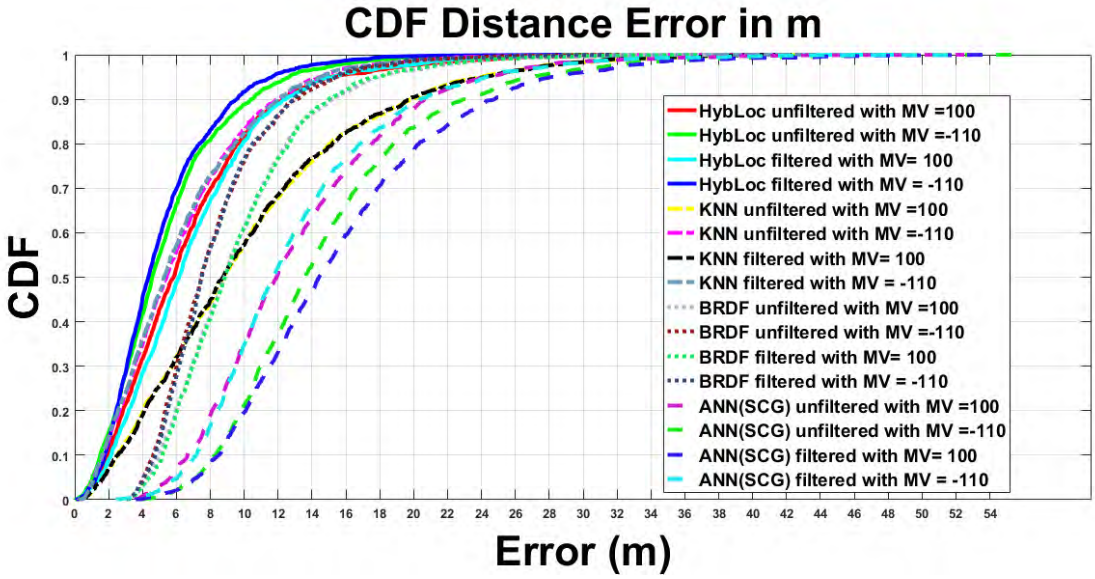

FIGURE 11. CDF of HybLoc vs kNN, Base-RF, and ANN (SCG) on building 0 stratified 30% unseen test data.

4-Layer ANN can take lesser time (ANN, 4-L, RBP, UnfltrdMVn110, 67.40 seconds) to converge than a 2-Layer network (ANN, 2-L, RBP, UnfltrdMVn110, 105.27 seconds) as indicated in Table 4 depending on several ANN parameters which govern the rate of convergence. Results on Building 1 are presented in Table. 6.

The similar trend is observed on Building 1, where on filtered dataset, the performance measures were slightly improved in comparison with their unfiltered counterparts for all approaches with both missing values in terms of accuracy, precision and recall. Particularly precision and recall were improved for all versions of filtered datasets except for ANN (RBP). In all four cases, HybLoc showed significantly better performance than all other IPS. For both Unfltrd and Fltrd datasets, missing value −110 dBm resulted in improved accuracy for SCG and decreased accuracy for RBP. Filtration of data with both missing values resulted in overall performance enhancement in case of training algorithm SCG as well as RBP. Training and response time for building 1 are presented in Table. 7.

Training time on building 1 remained almost unchanged with missing value variation. In case of filtered dataset, there was a slight reduction in training time mostly because of number of samples reduction in the filtered dataset. The response time by kNN for building 1 was similar to building 0 i.e. the response time remained practically the same,

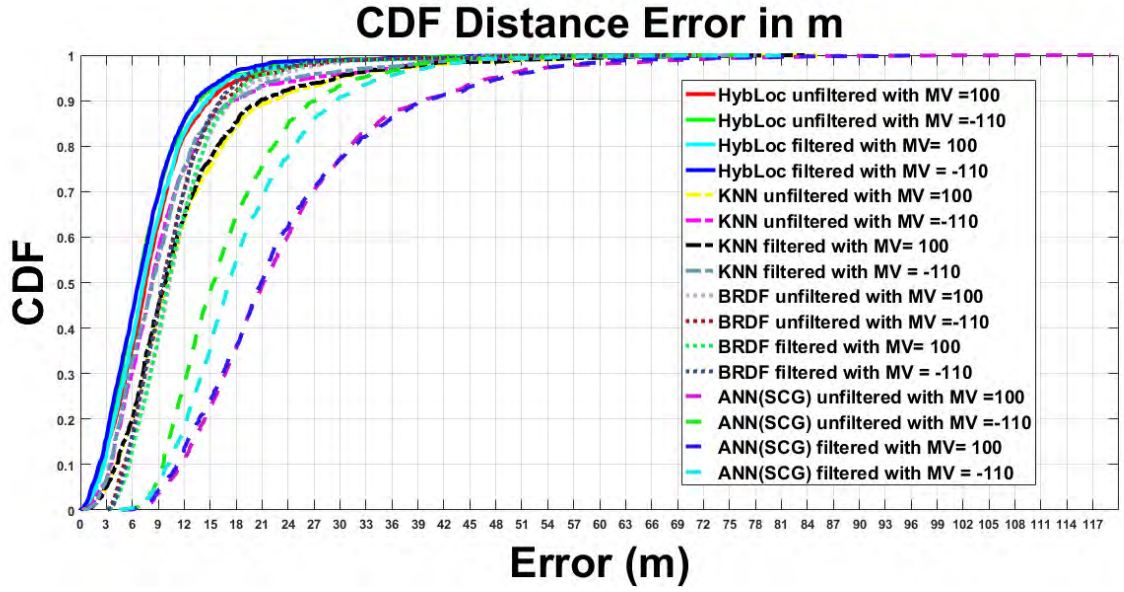

FIGURE 12. CDF of HybLoc vs kNN, Base-RF, and ANN (SCG) on building 1 stratified 30% unseen test data.

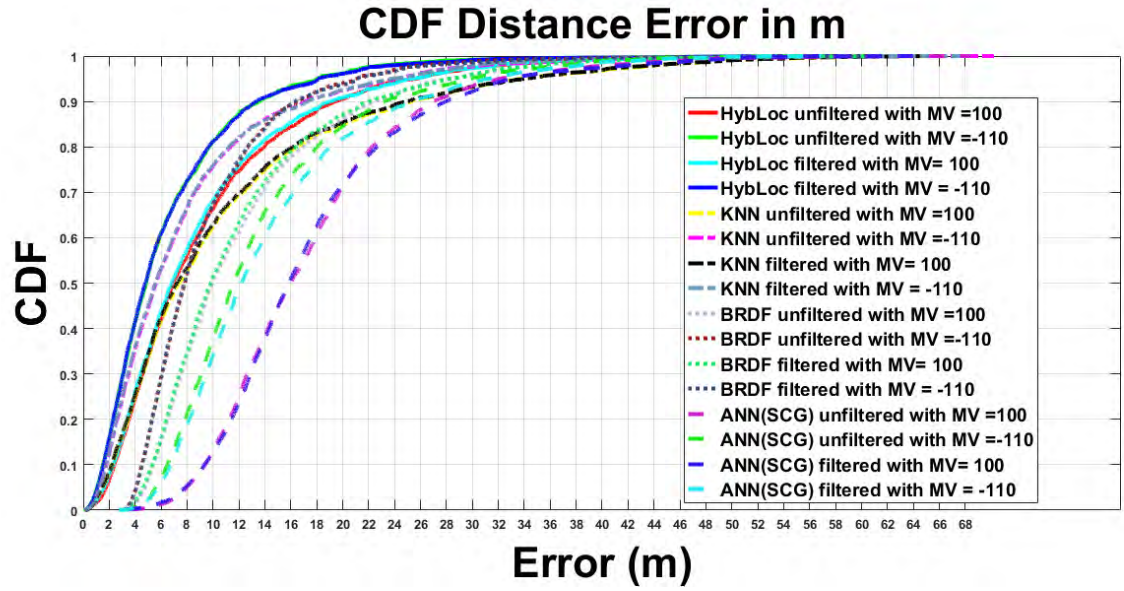

FIGURE 13. CDF of HybLoc vs kNN, Base-RF, and ANN (SCG) on building 2 stratified 30% unseen test data.

but the trend shown by HybLoc was exactly the opposite from the one shown for building 0. Earlier response time reduction of 10 times was observed with both missing value −110dBm used as well as filtered dataset case. However, for building 1 response time increased 10 times in case of missing value −110dBm instead of +100dBm and also increased 100 times with filtered dataset in comparison with unfiltered dataset counterpart. In case of building 1, 30 rooms out of 162 were filtered based on sample density. The overall number of samples for building 0 and specially building 2 are far greater than number of samples for building 1 after data filtration. The ANN response time (again on scale of E-05 seconds) was the fastest and remained consistent with the outcomes from building 0. The training time was reduced on Fltrd dataset because of lesser number of samples. Missing value impact on training time did not follow any specific pattern, at times decreasing with MVn110 and sometimes increasing. Base-RF response time remained persistent on scale of E-03 seconds however the response time of HybLoc fluctuates depending on how many RDF ensembles are invoked at run time based on soft cluster membership determination.

TABLE 12. Building 2 latitude-longitude level positioning error in meter.

IPS	Dataset	Min	Max	Mean	Mode	Std
HybLoc	UnfltrdMV100	0.18	57.99	9.36	4.91	7.72
kNN	UnfltrdMV100	0.13	66.21	11.02	5.40	10.53
Base-RF	UnfltrdMV100	3.18	60.99	12.36	15.79	7.72
ANN, 2-L, SCG	UnfltrdMV100	0.24	112.79	20.28	4.97	14.25
ANN, 2-L, RBP	UnfltrdMV100	1.13	633.32	156.41	63.51	103.61
ANN, 3-L, SCG	UnfltrdMV100	0.23	116.42	19.73	5.71	13.98
ANN, 3-L, RBP	UnfltrdMV100	13.97	702.11	321.99	283.94	104.60
ANN, 4-L, SCG	UnfltrdMV100	0.11	68.74	12.41	3.85	8.97
ANN, 4-L, RBP	UnfltrdMV100	124.93	799.25	365.63	398.98	113.34
HybLoc	UnfltrdMVn110	0.01	57.63	6.56	4.86	5.95
kNN	UnfltrdMVn110	0.08	70.21	7.96	5.40	7.73
Base-RF	UnfltrdMVn110	3.01	60.63	9.56	15.72	5.95
ANN, 2-L, SCG	UnfltrdMVn110	0.23	118.25	15.36	6.72	11.14
ANN, 2-L, RBP	UnfltrdMVn110	28.69	539.41	277.17	250.12	74.68
ANN, 3-L, SCG	UnfltrdMVn110	0.22	87.96	12.85	6.54	9.11
ANN, 3-L, RBP	UnfltrdMVn110	53.02	470.55	264.72	229.18	50.73
ANN, 4-L, SCG	UnfltrdMVn110	0.19	70.57	12.72	5.34	9.05
ANN, 4-L, RBP	UnfltrdMVn110	148.42	515.91	309.39	289.11	52.18
HybLoc	FltrdMV100	0.22	58.60	9.15	5.06	7.73
kNN	FltrdMV100	0.13	66.21	10.87	5.40	10.47
Base-RF	FltrdMV100	3.22	61.60	12.14	15.90	7.72
ANN, 2-L, SCG	FltrdMV100	0.27	138.61	23.83	5.12	16.53
ANN, 2-L, RBP	FltrdMV100	9.34	551.65	213.26	253.88	91.87
ANN, 3-L, SCG	FltrdMV100	0.23	73.18	13.68	5.89	9.84
ANN, 3-L, RBP	FltrdMV100	1.15	484.60	138.71	78.30	91.63
ANN, 4-L, SCG	FltrdMV100	0.32	92.45	15.24	5.50	10.55
ANN, 4-L, RBP	FltrdMV100	110.92	797.83	423.98	378.54	106.82
HybLoc	FltrdMVn110	0.14	58.16	6.59	5.00	6.06
kNN	FltrdMVn110	0.10	67.85	7.90	5.40	7.62
Base-RF	FltrdMVn110	3.14	61.16	9.58	15.87	6.06
ANN, 2-L, SCG	FltrdMVn110	0.42	103.79	16.67	7.13	12.52
ANN, 2-L, RBP	FltrdMVn110	203.57	985.19	640.22	667.52	92.59
ANN, 3-L, SCG	FltrdMVn110	0.20	74.53	13.38	6.79	9.50
ANN, 3-L, RBP	FltrdMVn110	124.56	705.39	293.63	330.32	52.23
ANN, 4-L, SCG	FltrdMVn110	0.27	81.69	13.19	8.06	9.38
ANN, 4-L, RBP	FltrdMVn110	429.29	769.69	615.45	584.31	47.21

Performance evaluation measure and training-response time for building 2 are presented in Table. 8 and 9 respectively.

For building 2 data, from Table. 8, it can be seen that the accuracy of HybLoc improved from 79% to 84% along with significant improvement in precision and recall in case of missing value changed to −110dBm. The same effect was observed with filtered dataset with both missing values −110dBm as well as +100dBm where accuracy changed from 79% to 84%. Over again HybLoc performed much better than kNN, Base-RF, and ANN based approach in all four cases, except for FltrdMV100 case, where the accuracy of both Base-RF and HybLoc was 0.79.

Training and response times for building 2 data from Table. 9, indicate that training time for HybLoc was slightly decreased with missing value −110dBm instead of +100dBm and also with filtered dataset and −110 dBm value, this training time reduction was observed. For building 2, HybLoc remained 10 times faster than Base-RF. kNN and ANN based approaches showed again similar response times of E-04 and E-05 seconds respectively. Although their response times are lesser than HybLoc's response time but HybLoc had a response time of E-03 seconds with significantly higher accuracy, precision and recall than kNN, ANN and Base-RF. Also response time variation of E-01 to E-03 seconds cannot be detected by any human utilizing the IPS. After detailing the results for each building individually, Fig. 5 and Fig. 6 depict pictorially the averaged overall trend of the performance measures for the complete dataset encompassing all buildings.

The overall mean performance measures also tally with the trends observed previously, as shown in Fig. 5 HybLoc showed overall significantly better performance than kNN, ANN and Base-RF in all four cases. When missing value +100dBm was replaced with −110dBm in unfiltered as well as filtered datasets, all IPS performed comparatively better except for ANN. The reason behind is that the training and tuning of ANN is not straightforward. There are many

TABLE 13. HybLoc training-testing, latitde-longitude level percentile error in meter on Building 0.

		Percentile			
IPS	Dataset	25^{th}	50^{th}	75^{th}	95^{th}
$Train_{HybLoc}$	UnfltrdMV100	3	5.1	7.6	13.5
$Test_{HybLoc}$	UnfltrdMV100	3.3	5.8	8.8	15.0
$Train_{HybLoc}$	UnfltrdMVn110	2.3	4.0	6.0	10.7
$Test_{HybLoc}$	UnfltrdMVn110	2.7	4.5	6.9	12.62
$Train_{HybLoc}$	FltrdMV100	3.3	5.5	8.1	12.9
$Test_{HybLoc}$	FltrdMV100	3.6	6.1	9.0	15.0
$Train_{HybLoc}$	FltrdMVn110	2.3	3.8	5.7	9.9
$Test_{HybLoc}$	FltrdMVn110	2.7	4.3	6.5	11.6

TABLE 14. HybLoc training-testing, latitde-longitude level percentile error in meter on Building 1.

		Percentile			
IPS	Dataset	25^{th}	50^{th}	75^{th}	95^{th}
$Train_{HybLoc}$	UnfltrdMV100	4.18	6.9	10.0	17.3
$Test_{HybLoc}$	UnfltrdMV100	4.76	7.5	10.6	19.3
$Train_{HybLoc}$	UnfltrdMVn110	3.8	6.2	9.1	16.1
$Test_{HybLoc}$	UnfltrdMVn110	4.1	6.7	9.8	16.7
$Train_{HybLoc}$	FltrdMV100	4.0	6.7	9.9	16.9
$Test_{HybLoc}$	FltrdMV100	4.5	7.2	10.5	17.4
$Train_{HybLoc}$	FltrdMVn110	3.5	6.1	9.0	15
$Test_{HybLoc}$	FltrdMVn110	3.9	6.7	9.7	16.6

TABLE 15. HybLoc training-testing, latitde-longitude level percentile error in meter on Building 2.

		Percentile			
IPS	Dataset	25th	50th	75th	95th
$Train_{HybLoc}$	UnfltrdMV100	3.3	5.8	10.4	21.5
$Test_{HybLoc}$	UnfltrdMV100	4.0	6.9	12.1	25.1
$Train_{HybLoc}$	UnfltrdMVn110	2.2	4.0	7.1	15.5
$Test_{HybLoc}$	UnfltrdMVn110	2.6	4.8	8.7	18.0
$Train_{HybLoc}$	FltrdMV100	3.2	5.6	10.1	21.1
$Test_{HybLoc}$	FltrdMV100	3.9	6.8	11.8	25.1
$Train_{HybLoc}$	FltrdMVn110	2.2	4.0	7.0	15.8
$Test_{HybLoc}$	FltrdMVn110	2.5	4.7	8.5	18.1

TABLE 16. Building 0 test dataset latitude-longitude level percentile error in meter.

		Percentile			
IPS	Dataset	25th	50th	75th	95th
HybLoc	UnfltrdMV100	3.3	5.8	8.8	15.0
kNN	UnfltrdMV100	4.8	8.9	13.6	23.9
Base-RF	UnfltrdMV100	6.5	8.8	11.84	17.9
ANN(SCG)	UnfltrdMV100	8.9	11.9	16.2	24.9
HybLoc	UnfltrdMVn110	2.7	4.5	6.9	12.62
kNN	UnfltrdMVn110	3.1	5.5	8.4	14.5
Base-RF	UnfltrdMVn110	5.7	7.4	9.7	15.5
ANN(SCG)	UnfltrdMVn110	10.1	13.5	17.7	27.1
HybLoc	FltrdMV100	3.6	3.6	6.1	9.0
kNN	FltrdMV100	4.8	8.7	13.4	24.3
Base-RF	FltrdMV100	6.5	8.8	11.8	18.0
ANN(SCG)	FltrdMV100	10.7	14.4	19.2	27.9
HybLoc	FltrdMVn110	2.7	4.3	6.5	11.6
kNN	FltrdMVn110	3.0	5.2	8.3	14.3
Base-RF	FltrdMVn110	5.8	7.4	9.7	15.5
ANN(SCG)	FltrdMVn110	8.9	11.7	15.3	24.1

generic guidelines for its design but no particular rules for a huge number of algorithmic parameters. Although we chose some common configurations averaged over 3 combinations as described earlier but the resulting performance measures were highly fluctuating hence affecting the overall mean. If we draw comparisons focusing filtered vs unfiltered dataset, then on filtered dataset all IPS performed better than unfiltered data which indicates that suitable missing value as well as sufficiently large number of samples collected per location play a significant role in overall performance of any IPS.

Fig. 6 sheds light on training and response times averaged over all buildings in the dataset. Log_{10} of training time (in seconds) was taken twice to make the value sufficiently smaller to be suitable for pictorial depiction along with response time which is simply given in seconds. Training time for kNN is Nil, the mean training time for HybLoc showed consistent drop starting from unfiltered with MV100, unfiltered with MVn110, filtered with MV100 and filtered with MVn110 respectively. This change is visible in graph as the time dropped from 0.30 to 0.25 seconds (log_{10} taken twice). The response time slightly dropped for unfiltered dataset with MV100 to MVn110. However, the response time was quite large for MVn110 than MV100 for filtered dataset. The major factor which increased both averaged training and response

TABLE 17. Building 1 test dataset latitude-longitude level percentile error in meter.

		Percentile			
IPS	Dataset	25th	50th	75th	95th
HybLoc	UnfltrdMV100	4.76	7.5	10.6	19.3
kNN	UnfltrdMV100	6.6	9.9	14.9	29.9
Base-RF	UnfltrdMV100	7.4	10.3	13.5	21.8
ANN(SCG)	UnfltrdMV100	15.7	21.1	29.2	46.9
HybLoc	UnfltrdMVn110	4.1	6.7	9.8	16.7
kNN	UnfltrdMVn110	5.2	8.2	12.0	27.1
Base-RF	UnfltrdMVn110	6.91	9.5	12.6	19.7
ANN(SCG)	UnfltrdMVn110	11.5	15.2	20.7	32.5
HybLoc	FltrdMV100	4.5	7.2	10.5	17.4
kNN	FltrdMV100	6.6	9.7	14.1	29.2
Base-RF	FltrdMV100	7.5	10.1	13.5	19.9
ANN(SCG)	FltrdMV100	15.3	20.8	29.2	47.9
HybLoc	FltrdMVn110	3.9	6.7	9.7	16.6
kNN	FltrdMVn110	5.1	8.3	11.9	24.1
Base-RF	FltrdMVn110	6.8	9.4	12.7	18.6
ANN(SCG)	FltrdMVn110	13.2	17.1	22.8	35.2

TABLE 18. Building 2 test dataset latitude-longitude level percentile error in meter.

		Percentile			
IPS	Dataset	25th	50th	75th	95th
HybLoc	UnfltrdMV100	4.0	6.9	12.1	25.1
kNN	UnfltrdMV100	3.9	7.4	14.0	33.7
Base-RF	UnfltrdMV100	7.0	9.9	15.1	28.6
ANN(SCG)	UnfltrdMV100	11.9	16.0	20.9	31.8
HybLoc	UnfltrdMVn110	2.6	4.8	8.7	18.0
kNN	UnfltrdMVn110	3.1	5.6	9.8	24.1
Base-RF	UnfltrdMVn110	5.6	7.8	11.7	21.0
ANN(SCG)	UnfltrdMVn110	8.4	11.6	16.3	30.8
HybLoc	FltrdMV100	3.9	6.8	11.8	25.1
kNN	FltrdMV100	4.0	7.2	13.9	33.6
Base-RF	FltrdMV100	6.9	9.8	14.7	28
ANN(SCG)	FltrdMV100	12.3	15.7	20.9	33.8
HybLoc	FltrdMVn110	2.5	4.7	8.5	18.1
kNN	FltrdMVn110	3.0	5.6	9.7	23.5
Base-RF	FltrdMVn110	5.6	7.6	11.5	21.1
ANN(SCG)	FltrdMVn110	8.8	12.2	17.3	32.3

times for HybLoc was due to the performance in building 1. This building had 163 rooms in it but the overall number of samples per room were not high, mostly looming slightly over the minimum threshold of samples. Fewer number of samples per location in building 1 resulted in more complex converged model of the IPS HybLoc, resulting in increased training as well response time. Training time for Base-RF and ANN remained almost same MV100 and MVn110. However, it reduced a little for Fltrd datasets with both missing values. Response time for both Base-RF and ANN was minimal which remained consistent for all 4 combinations.

B. LATITUDE-LONGITUDE PREDICTION RESULTS

The results for latitude-longitude prediction were obtained through the same pipeline of GMM based soft clustering and I out of N RDF ensembles invocation based on minimum threshold for cluster membership determination. The major difference here was the use of regression ensembles instead of classification. For kNN, the implementation was modified

to produce the mean of the matched k nearest neighbors' latitude values as well as longitude values for generating the final output of latitude and longitude respectively. Base-RF results were generated with direct application of Random Forest per building dataset with exactly same parameters used for HybLoc i.e. 300 trees, 25 random features, and 1,024 maximum splits per tree, one such ensemble was trained for each latitude and longitude prediction. The 2, 3, and 4-Layer ANN were trained with same configuration for both for latitude and longitude with training algorithm SCG and RBP. The resultant latitude and longitude values were then used as predicted position which was compared with ground truth latitude-longitude values pair to compute Euclidean distance based positioning error. The following results for all 3 individual buildings were generated using the same aforementioned strategy. It must be noted that the results presented in this section were computed with unseen 30% stratified test dataset for each building. The performance measures for building 0 are shown in Table. 10.

It can be seen from Table. 10, that for regression/ latitude-longitude prediction missing value −110dBm was found to be providing better performance in comparison with +100dBm for HybLoc and other IPS. For missing value +100dBm, comparing Unfltrd and Fltrd dataset, performance of HybLoc degraded but for MVn110 comparing the same, its performance was slightly improved considering the mean error reduced from 5.42m to 5.13m. Results for building 1 and 2 are expressed in Table. 11 and 12 respectively.

For building 1, the impact of MVn110 and filtration of dataset is clearly visible from Table. 11 in form of overall performance improvement. Pairwise unfiltered and filtered all four cases, as well as for missing value changed to −110dBm in both cases is in accordance with findings from room-level results that data filtration as well as −110dBm missing value improved the system performance.

The results on building 2 depicted in Table. 12, shows performance in case of missing value changed from +100 to −110 dBm but if Unfltrd and Fltrd cases are compared pairwise then a trivial performance degradation is observed rather than any improvement. The averaged positioning error over complete dataset is provided in Fig. 7 for quick visual comparison. It can be deduced that both missing value replacement of −110dBm instead of +100dBm and data filtration were found useful for room-level prediction performance enhancement. However, for latitude-longitude level prediction usage of −110dBm missing value instead of +100dBm was found useful but data filtration did not bring as significant performance improvement as in its room-level prediction counterpart.

The minimum error obtained by HybLoc, kNN, Base-RF, and ANN for UnfltrdMV100, UnfltrdMVn110, FltrdMV100 and FltrdMVn110 were (0.16, 0.26, 3.17, 32.9), (0.06, 0.26, 3.07, 81.64), (0.22, 0.30, 3.23, 17.7), and (0.08, 0.26, 3.12, 80.99) m respectively. These results on positioning error in meters do provide some useful insight but the true picture becomes clear with help of CDF which provides a holistic view of the performance of IPS for all tested samples. On all buildings, the ANN configurations with training algorithm RBP provided far worse results than SCG. It remained valid for all 2-Layer, 3-Layer and 4-Layer ANN configurations. Hence the CDF of ANN with RBP used as training algorithm are provided as supplementary material for the interested reader but are not included in Fig. 11-13.

First, the results are reported building-wise on both training and testing data indicating that in case of small dataset 10-fold cross validation can also provide meaningful insights on the performance of data. Secondly, the performance of HybLoc is compared with kNN, Base-RF (to validate HybLoc advantage over straight forward application of Random Forest), and ANN, the most popular machine learning techniques frequently used for indoor localization.

In Fig. 8, 10-fold cross validated (10-CV) results on training data as well as results on 30% unseen test data are provided for aforementioned four cases from Section V. Eighty percent of the training set 10-CV results showed positioning error under 8.3 m for UnfltrdMV100 whereas for same results on test data indicate that 80% of the tested samples generated positioning error under 9.6m. The same training and test data kept Unfltrd but with MV = −110dBm produced less than or up to 6.7m error in 80% of the training data and for testing data, it was 7.7m. For Fltrd dataset with MV=100dBm, 80% of the training data and testing data produced positioning error of 8.8m and 9.9m respectively. The best performance was shown by Fltrd datasets with MV

TABLE 19. Performance comparison with related work on same dataset.

IPS Ref. No.	Room-Level Prediction				Lat-Long Prediction			
	Accuracy	Precision	Recall	1 FP Response Time(sec)	Min. Error	Max. Error	Mean Error	1 FP Response Time(sec)
[18]	-	-	-	-	4.73	-	-	9.78E-03
[23]	-	-	-	-	8.21	-	-	-
[30]	0.85	-	-	-	-	-	-	-
[31]	-	-	-	-	-	-	RMSE:12.21 m for long., 10.12 for lat.	-
[32]	0.78	-	-	-	-	-	-	-
HybLoc	0.85	0.89	0.83	7.21E-01	0.08	52.17	6.46	7.21E-01

−110dBm with 80% of the samples showing error bounded by 6.3m for training and 7.3m for testing data.

The results for building 0 are summarized in Table. 13 for 10-CV 70% training data and unseen 30% test data in terms of 25th, 50th, 75th, and 95th percentile of positioning error.

The CDF for building 1 is shown in Fig. 9 with training and testing data results on all four cases of missing value and dataset filtration.

The summarized results reporting 25, 50, 75, and 95 percent of the samples' bounded error in meters is presented in Table. 14.

Following the same pattern, the CDF for building 2 including results on training and testing data are expressed in Fig. 10. The summary of results in terms of 25th, 50th, 75th, and 95th percentile is presented in Table. 15.

The overall trend from all three buildings positioning error shows that 10-fold CV results and results obtained on unseen test data are quite close with approximately 1 to 2 m difference in every case individually. Moreover, the use of appropriate missing value can be a major factor to influence the IPS performance, missing value −110dBm was found to be consistently better than +100dBm throughout for all three buildings. Sufficiently large number of samples per location of interest was also helpful for the IPS to distinguish different places more efficiently as evident by filtered dataset's performance being better than its unfiltered counterpart in majority of all four cases.

The results are presented now in terms of CDFs of the HybLoc and compared IPS for building 0, building 1, and building 2 in Fig. 11, Fig.12, and Fig. 13 respectively.

The results detailed in Fig.11 are for building 0, 30% stratified unseen data, on which performance of HybLoc and other IPS are compared. The results for kNN presented were averaged over 6 different configurations whereas ANN (SCG) and ANN (RBP) results were computed using 3 different configurations for each 2, 3 and 4-Layer networks whose mean values are reported. The summarized results for building 0 are expressed in Table. 16.

On building 0 test data, the 25th percentile remained almost the same for Unfltrd and Fltrd pairwise parts, overall HybLoc performing better than other approaches and missing value −110dBm being better than 100. However, for 50th percentile onwards, Fltrd datasets produced better than results than their pairwise Unfltrd counterparts.

For building 1, the performance of HybLoc for missing value −110 was better than 100 in both Unfltrd and Fltrd cases. The performance for both Unfltrd and Fltrd data using same missing value was almost same as seen in Fig. 12 indicated by very close CDFs. The summarized positioning error on building 1 is presented in Table. 17.

For building 1, the same trend is observed for 25th, 50th, 75th, and 95th percentile with missing value −110dBm producing far less positioning errors for both Unfltrd and Fltrd datasets. Moreover, HybLoc clearly outperformed other approaches.

Results for building 2 are shown in Fig.13 along with percentile positioning error summary in Table. 18.

On building 2 data, again HybLoc and kNN remained a close match in 25th percentile case with visible performance improvement from 50th percentile onwards.

The summarized performance comparison of HybLoc with related work on same dataset is presented in Table 19.

It is evident from Table 19 that majority of the work utilizing UJIIndoorLoc dataset either report results for room-level prediction or latitude-longitude prediction. HybLoc not only provides results for both, but the performance in terms of accuracy, minimum error, and mean error is better than all related work except [30] where the accuracy is 85% by both IPS. However, it should be noted that HybLoc provides detailed performance measures than merely accuracy on complete data of all buildings and [30] provided accuracy results on only few selected regions instead of whole dataset. HybLoc provided accuracy of 85% on all 3 buildings data rather than a small number of regions/rooms.

VI. CONCLUSION

In this work, we proposed a new hybrid indoor Wi-Fi localization system based on Random Decision Forest ensembles utilizing GMM soft clustering for dataset partitioning. Ensemble methods combine strength of many weak learners to improve the overall accuracy as well as generalization capability which is very important in real world Wi-Fi fingerprinting based indoor location prediction. Our system extended the idea of combining weak learners to generate ensemble of ensembles. Data partitioning based on soft clustering enables the inclusion of relevant samples in training of RDF ensembles, at the same time dividing the dataset to enable numerous classifiers and regression models learn the partitioned dataset structure rather enforcing a single one to learn the complete diverse dataset. The localization results were presented on both room-level as well as latitude-longitude level prediction to allow comparison of two major localization streams in the literature. We used a publically available, large Wi-Fi fingerprints database UJIIndoorLoc instead of a proprietary small lab/floor level dataset, allowing the reader to directly compare many existing works in the Wi-Fi based localization. We further extended the experiments to explore and identify the impact of missing value replacement in the Wi-Fi fingerprints along with impact of sufficiently large number of fingerprint samples per location for performance improvement. The experiments demonstrated that the proposed system is featured with high localization accuracy with response time suitable for real-world practical applications requiring either room-level or coordinate level location estimation.

ACKNOWLEDGMENT

The authors would like to acknowledge the technical and administrative support of Dr. Usman Ghani Khan and Dr. Hafiz Shahzad Asif, Department of Computer Science and Engineering, University of Engineering and Technology, Lahore, Pakistan, for this work.

REFERENCES

[1] R. Berkvens, H. Peremans, and M. Weyn, "Conditional entropy and location error in indoor localization using probabilistic Wi-Fi fingerprinting," *Sensors*, vol. 16, no. 10, pp. 1–21, 2016.

[2] X. Wang, L. Gao, and S. Mao, "BiLoc: Bi-modal deep learning for indoor localization with commodity 5 GHz WiFi," *IEEE Access*, vol. 5, pp. 4209–4220, 2017.

[3] H. Shin, Y. Chon, Y. Kim, and H. Cha, "MRI: Model-based radio interpolation for indoor war-walking," *IEEE Trans. Mobile Comput.*, vol. 14, no. 6, pp. 1231–1244, Jun. 2015.

[4] S. Li, Y. Lou, and B. Liu, "Bluetooth aided mobile phone localization: A nonlinear neural circuit approach," *ACM Trans. Embedded Comput. Syst.*, vol. 13, no. 4, p. 78, 2014.

[5] J. T. Biehl, A. J. Lee, G. Filby, and M. Cooper, "You're where? Prove it!: Towards trusted indoor location estimation of mobile devices," in *Proc. ACM Int. Joint. Conf. Pervasive Ubiquitous Comput.*, 2015, pp. 909–919.

[6] C. Xiao, D. Yang, Z. Chen, and G. Tan, "3-D BLE indoor localization based on denoising autoencoder," *IEEE Access*, vol. 5, pp. 12751–12760, 2017.

[7] L. Calderoni, M. Ferrara, A. Franco, and D. Maio, "Indoor localization in a hospital environment using random forest classifiers," *Expert Syst. Appl.*, vol. 42, no. 1, pp. 125–134, 2015.

[8] A. Aguilar-Garcia, S. Fortes, E. Colin, and R. Barco, "Enhancing RFID indoor localization with cellular technologies," *EURASIP J. Wireless Commun. Netw.*, vol. 2015, no. 1, p. 219, Dec. 2015.

[9] J. Luo and H. Gao, "Deep belief networks for fingerprinting indoor localization using ultrawideband technology," *Int. J. Distrib. Sens. Netw.*, vol. 12, no. 1, p. 5840916, 2016.

[10] Y. Sun, W. Meng, C. Li, N. Zhao, K. Zhao, and N. Zhang, "Human localization using multi-source heterogeneous data in indoor environments," *IEEE Access*, vol. 5, pp. 812–822, 2017.

[11] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "SpotFi: Decimeter level localization using WiFi," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 4, pp. 269–282, 2015.

[12] X. Wang, X. Wang, and S. Mao, "CiFi: Deep convolutional neural networks for indoor localization with 5 GHz Wi-Fi," in *Proc. IEEE Int. Conf. Commun.*, May 2017, pp. 1–6.

[13] W. Kang and Y. Han, "SmartPDR: Smartphone-based pedestrian dead reckoning for indoor localization," *IEEE Sensors J.*, vol. 15, no. 5, pp. 2906–2916, May 2015.

[14] P. Cremonese, D. Gallucci, M. Papandrea, S. Vanini, and S. Giordano, "PROMO: Continuous localized and profiled multimedia content distribution," in *Proc. 3rd Workshop Mobile Video Del.*, 2010, pp. 21–26.

[15] R. Górak and M. Luckner, "Malfunction immune Wi–Fi localisation method," in *Computational Collective Intelligence*. Cham, Switzerland: Springer, 2015, pp. 328–337.

[16] Ó. Belmonte-Fernández, A. Puertas-Cabedo, J. Torres-Sospedra, R. Montoliu-Colás, and S. Trilles-Oliver, "An indoor positioning system based on wearables for ambient-assisted living," *Sensors*, vol. 17, no. 1, pp. 1–22, 2017.

[17] A. H. Salamah, M. Tamazin, M. A. Sharkas, and M. Khedr, "An enhanced WiFi indoor localization system based on machine learning," in *Proc. Int. Conf. Indoor Position. Indoor Navigat.*, Oct. 2016, pp. 1–8.

[18] J. Torres-Sospedra, R. Montoliu, G. M. Mendoza-Silva, O. Belmonte, D. Rambla, and J. Huerta, "Providing databases for different indoor positioning technologies: Pros and cons of magnetic field and Wi-Fi based positioning," *Mobile Inf. Syst.*, vol. 2016, Apr. 2016, Art. no. 6092618, doi: 10.1155/2016/6092618.

[19] M. Zhou, Y. Wei, Z. Tian, X. Yang, and L. Li, "Achieving cost-efficient indoor fingerprint localization on WLAN platform: A hypothetical test approach," *IEEE Access*, vol. 5, pp. 15865–15874, 2017.

[20] N. Li, J. Chen, Y. Yuan, X. Tian, Y. Han, and M. Xia, "A Wi-Fi indoor localization strategy using particle swarm optimization based artificial neural networks," *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 3, p. 33, 2016.

[21] C. Song, J. Wang, and G. Yuan, "Hidden naive Bayes indoor fingerprinting localization based on best-discriminating AP selection," *ISPRS Int. J. Geo-Inf.*, vol. 5, no. 10, p. 189, 2016.

[22] G. Ding, Z. Tan, J. Zhang, and L. Zhang, "Fingerprinting localization based on affinity propagation clustering and artificial neural networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2013, pp. 2317–2322.

[23] J. Wietrzykowski, M. Nowicki, and P. Skrzypczyński, "Adopting the FAB-MAP algorithm for indoor localization with WiFi fingerprints," in *Proc. Int. Conf. Automat.*, vol. 550, Mar. 2017, pp. 585–594.

[24] M. Bernas and B. Płaczek, "Fully connected neural networks ensemble with signal strength clustering for indoor localization in wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 12, p. 403242, 2015.

[25] R. Górak and M. Luckner, "Modified random forest algorithm for Wi–Fi indoor localization system," in *Proc. Int. Conf. Comput. Collective Intell.* Cham, Switzerland: Springer, 2016, pp. 147–157.

[26] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. IEEE Conf. Comput. Commun., 19th Annu. Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, vol. 2, Mar. 2000, pp. 775–784.

[27] L. Kanaris, A. Kokkinis, A. Liotta, and S. Stavrou, "Combining smart lighting and radio fingerprinting for improved indoor localization," in *Proc. IEEE 14th Int. Conf. Netw., Sens. Control. (ICNSC)*, May 2017, pp. 447–452.

[28] M. Cooper, J. Biehl, G. Filby, and S. Kratz, "LoCo: Boosting for indoor location classification combining Wi-Fi and BLE," *Pers. Ubiquitous Comput.*, vol. 20, no. 1, pp. 83–96, 2016.

[29] P. Bolliger, "Redpin—Adaptive, zero-configuration indoor localization through user collaboration," in *Proc. 1st ACM Int. Workshop Mobile Entity Localization Tracking GPS-Less Environ. (MELT)*, vol. 8, 2008, pp. 55–60.

[30] S. Bozkurt, G. Elibol, S. Gunal, and U. Yayan, "A comparative study on machine learning algorithms for indoor positioning," in *Proc. Int. Symp. Innov. Intell. Syst. Appl.*, Sep. 2015, pp. 1–8.

[31] M. T. Uddin and M. M. Islam, "Extremely randomized trees for Wi-Fi fingerprint-based indoor positioning," in *Proc. 18th Int. Conf. Comput. Inf. Technol. (ICCIT)*, Dec. 2015, pp. 105–110.

[32] M. Nowicki and J. Wietrzykowski, "Low-effort place recognition with WiFi fingerprints using deep learning," *Adv. Intell. Syst. Comput.*, vol. 550, pp. 575–584, Mar. 2017.

[33] J. Torres-Sospedra *et al.*, "UJIIndoorLoc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Oct. 2014, pp. 261–270.

[34] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[35] C. Sansone, J. Kittler, and F. Roli, Eds., *Multiple Classifier Systems: 10th International Workshop, MCS 2011, Naples, Italy, June 15-17, 2011. Proceedings*, vol. 6713. Springer, 2011.

[36] T. Chakraborty, "EC3: Combining clustering and classification for ensemble learning," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2017, pp. 781–786.

[37] X. Ma, P. Luo, F. Zhuang, Q. He, Z. Shi, and Z. Shen, "Combining supervised and unsupervised models via unconstrained probabilistic embedding," in *Proc. 22nd Int. Joint Conf. Artif. Intell.*, 2011, pp. 1396–1401.

[38] J. Gao, F. Liang, W. Fan, Y. Sun, and J. Han, "A graph-based consensus maximization approach for combining multiple supervised and unsupervised models," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 1, pp. 15–28, Jan. 2013.

[39] P. S. Nagpal and R. Rashidzadeh, "Indoor positioning using magnetic compass and accelerometer of smartphones," in *Proc. Int. Conf. Sel. Topics Mobile Wireless Netw.*, Aug. 2013, pp. 140–145.

[40] K. Kaji and N. Kawaguchi, "Design and implementation of WiFi indoor localization based on Gaussian mixture model and particle filter," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Nov. 2012, pp. 1–9.

[41] L. Luoh, "ZigBee-based intelligent indoor positioning system soft computing," *Soft Comput.*, vol. 18, no. 3, pp. 443–456, 2014.

[42] Z. Zheng *et al.*, "BigLoc: A two-stage positioning method for large indoor space," *Int. J. Distrib. Sens. Netw.*, vol. 12, no. 6, p. 1289013, 2016.

BEENISH A. AKRAM was born in Lahore, Pakistan, in 1984. She received the B.Sc. degree (Hons.) in computer engineering and the M.Sc. degree in computer science from the University of Engineering and Technology (UET) at Lahore, Lahore, in 2006 and 2010, respectively, where she is currently pursuing the Ph.D. degree in computer science.

From 2006 to 2007, she was a Software Design Engineer with MicroTech Industries (Pvt.) Ltd., Lahore. From 2007 to 2012, she was a Lecturer with UET Lahore. Since 2012, she has been an Assistant Professor with the Department of Computer Science and Engineering, UET Lahore. Her research interests include machine learning, IoT, and indoor localization and embedded systems.

ALI H. AKBAR received the bachelor's degree (Hons.) in telecommunications from NUST, Pakistan, in 1997, the M.S. degree from UNSW Australia in 1999, and the Ph.D. degree from Ajou University in 2008. He is currently an Associate Professor with the University of Engineering and Technology at Lahore, Lahore, Pakistan. His topics of interest include wireless networks, such as MANETs and sensor network, M2M networks, and distributed systems. He was a consultant with the Al-Khawarizmi Institute of Computer Science for government organizations, such as Lahore Transport Company and RESCUE 1122.

Dr. Akbar was declared as a star laureate by South Asia Publications for the year of 2003. In 1998, he received the merit scholarship for the M.S. degree.

OMAIR SHAFIQ received the Ph.D. degree in computer science from the University of Calgary, Calgary, AB, Canada. He is currently an Assistant Professor with Carleton University, Ottawa, ON, Canada. His research interests include data modeling, big data analytics, services computing, machine learning, and cloud computing. He has been an Assistant Professor with the School of Information Technology, Carleton University. He has published over 60 peer-reviewed publications in journals, book chapters, conferences, and workshops, served in technical program committee of over 30 conferences and workshops, and co-organized over eight conference and workshops.

Dr. Shafiq was a recipient of the Departmental Research Award from the University of Calgary in 2009 and 2010, the Alberta-Innovates Technology Futures Scholarships for the master's and Ph.D. studies in 2010 and 2011, the Teaching Excellence Award from the University of Calgary in 2011, the NSERC Vanier CGS Scholarship in 2012, the J.B. Hyne Research Innovation Award from the University of Calgary in 2012, the NSERC Post-Doctoral Fellowship Award, and the Mitacs Elevate Postdoctoral Fellowship Award Competition in 2015–2016.