Mehran University Research Journal of Engineering & Technology
Vol. 38, No. 1, 151-174, January 2019
p-ISSN: 0254-7821, e-ISSN: 2413-7219
DOI: 10.22581/muet1982.1901.13

Wi-Fi Fingerprinting Based Room Level Indoor Localization Framework Using Ensemble Classifiers

BEENISH AYESHA AKRAM*, AND ALI HAMMAD AKBAR*

RECEIVED ON 22.12.2017 ACCEPTED ON 12.02.2018

ABSTRACT

Over the past decennium, Wi-Fi fingerprinting based indoor localization has seized substantial attention. Room level indoor localization can enable numerous applications to increase their diversity by incorporating user location. Real-world commercial scale deployments have not been realized because of difficulty in capturing radio propagation models. In case of fingerprinting based approaches, radio propagation model is implicitly integrated in the gathered fingerprints providing more realistic information as compared to empirical propagation models. We propose ensemble classifiers based indoor localization using Wi-Fi fingerprints for room level prediction. The major advantages of the proposed framework are, ease of training, ease to set up framework providing high room-level accuracy with trifling response time making it viable and appropriate for real time applications. It performs well in comparison with recurrently used ANN (Artificial Neural Network) and kNN (k-Nearest Neighbours) based solutions. Experiments performed showed that on our real-world Wi-Fi fingerprint dataset, our proposed approach achieved 89% accuracy whereas neural network and kNN based best found configurations achieved 85 and 82% accuracy respectively.

Key Words: Indoor Localization, Random Forest, Ensemble Classifiers, Indoor Positioning System.

1. INTRODUCTION

LAS (Location Aware Services) such as smart homes, personal assistance, smart cities, navigation and tracking of IoT (Internet of Things) objects has automated location identification as its core. The accuracy, precision and response time of localization framework determines the quality of these LAS.

Outdoor localization systems such as GPS (Global Positioning System), GLONAS (Global Navigation Satellite System), COMPASS, IRNSS (Indian Regional Navigation Satellite System), BAIDU, DORIS (Doppler Orbitography and Radio Positioning Integrated by Satellite) and GALLILEO cannot go beyond the building level granularity. Such systems are unsuitable for indoor

Authors E-Mail: (beenish.ayesha.akram@uet.edu.pk, ahakbar@gmail.com)
* Department of Computer Science & Engineering, University of Engineering & Technology, Lahore, Pakistan.

environments because of No-Line-of-Sight availability inside buildings. Whereas RF (Radio Frequency) signals easily bend to enter indoor locations.

RSSI (Received Signal Strength Indicator)values of RF signals such as Bluetooth, Wi-Fi, RFID (Radio-Frequency Identification) [1], ZigBee [2] and ultra wideband [3] have found place in many solutions proposed for indoor localization. Apart from RF signals, diverse types of sensory inputs such as video, ambient sound [4], magnetometer [5], gyroscope readings, pictures [6], accelerometer [7], and their combination with numerous aforementioned RF signals [8-9] have also been used for indoor localization.

Indoor localization techniques including PDR (Pedestrian Dead Reckoning), TOA (Time of Arrival) [10], TDOA (Time Difference of Arrival) [11-12], AOA (Angle of Arrival) and RSSI [13] have many shortcomings. PDR has the disadvantage of error propagation in successive estimates, whereas TOA and TDOA require accurate clock synchronization between sender and receiver. AOA demands specific antennas to be present in device and henceforth involves specialized hardware for localization.

With the popularity of smart phones, Bluetooth and Wi-Fi are noticeable in these works as they have become a de facto sensor on every smart device. Moreover, almost every civil infrastructure including homes, offices, educational institutions, airports etc. have pre-deployed Wi-Fi access points. This wavers off requirement of any additional equipment. Between both Wi-Fi and Bluetooth, Bluetooth requires explicitly a constellation of Bluetooth beacons such as iBeacon from Apple Inc.

Wi-Fi fingerprint based localization has the following benefits: extra hardware is not required neither at AP (Access Point) nor at smart device by using already existent infrastructure; straightforwardly implementable; empirical propagation model is not mandatory to build which may or may not depict real signal propagation during real world deployment [14].

Fingerprinting based framework include training phase and prediction phase which are also mentioned as Offline stage and Online stage in localization works respectively. During the training phase, the building area is usually partitioned into square cells and each RP (Reference Point) is marked at the centre of each such cell. Wi-Fi fingerprint (Radio Signal Signature: FP) is captured at each RP using appropriate sensory device and stored in database with marked known location which serves as the ground truth label. Usually numerous FPs are collected on each RP to handle the radio signal fluctuation. The collection of FPs along with ground truth location labels is optionally pre-processed before storage in database. Afterwards, the location database is used for training localization engine. During prediction phase, Radio signal FP at the anonymous location is taken and estimated location is calculated using the already trained localization engine in the training phase. The complete scenario of training and testing phase is pictorially explained in Fig. 1.

Majority of the fingerprinting based solutions face subsequent main problems. Firstly, the solutions based on kNN and its modifications do not scale well regarding response time ($t_{LocationComputed}$-$t_{LocationQuery}$) with increasing FP database size due to comparison with all the samples in dataset. Secondly, trained models have generalization problem because of overfitting which leads to low accuracy and precision predominantly at run time [15].We propose use of bootstrapped aggregated/bagged ensemble of decision trees also known as RDF (Random Decision Forest) for indoor localization. We intend to accomplish scalable, easily deployable, real-time response targeting high room level accuracy instead of (x,y) coordinates. The primary idea behind RF (Random Forest) is to build several decision trees in training phase by randomly selecting dataset predictors and random samples with replacement for splitting nodes and fitting each decision tree. The combination of bootstrap aggregation and implicit random predictor selection during training enables RDF to avoid overfitting and overtraining which results in enhanced generalization ability. Time required for training as well as prediction computation of RDF Ensemble also make it well-suited for nearly real time realization of localization system.

A custom build Android app was used to collect the Wi-Fi dataset and the proposed localization framework was implemented using MATLAB R2016a to evaluate the performance. The location dataset of SE (Software Engineering) Centre, Department of Computer Science & Engineering, UET (University of Engineering & Technology), Lahore, Pakistan, was used for the experimentation. Data was collected over several weeks on different times during which no specific constraints on pedestrians or normal departmental activities were imposed. Hence the results presented in this work describe real world scenario.

The manuscript is organized as follows. Section 2 deliberates related works regarding indoor localization based on radio frequency signals primarily Wi-Fi. Section 3 provides specifics of the proposed localization framework. Section 4 dwells into experimental setup, blueprint and outcomes to validate the proposed framework. Section 5 presents discussion and future work.

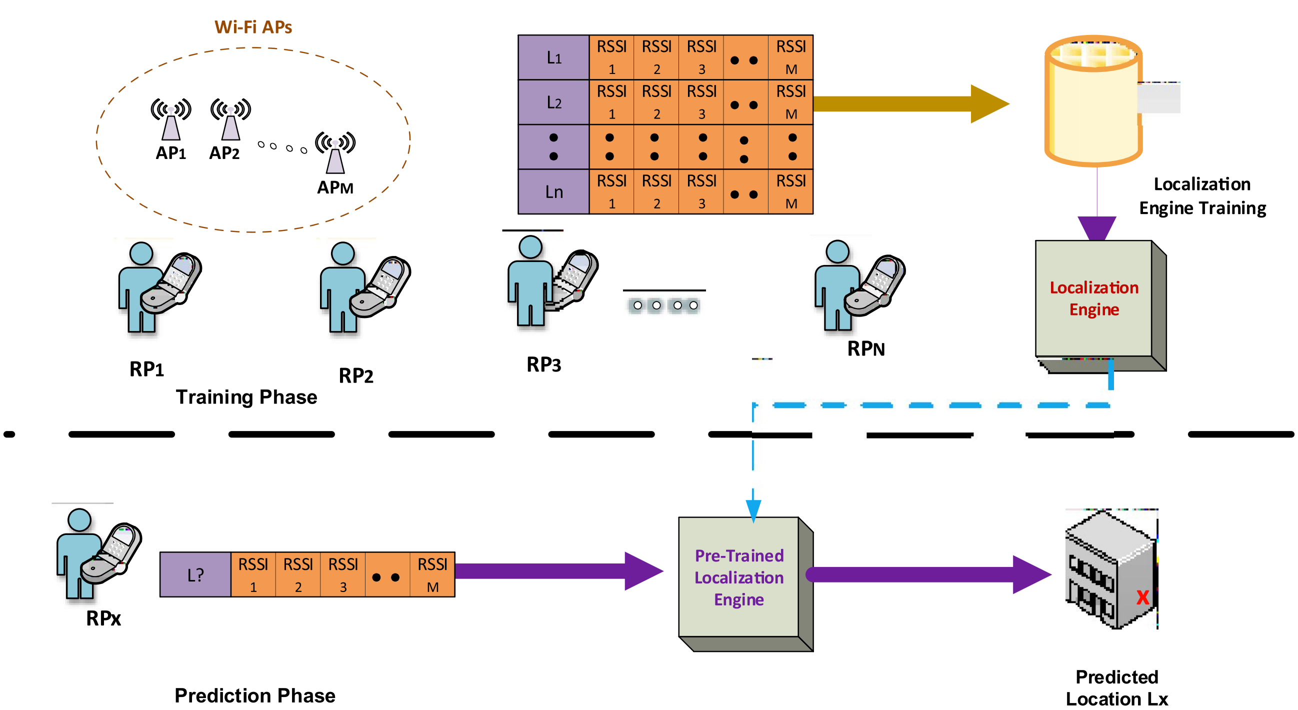

FIG. 1. GENERAL RADIO FREQUENCY FINGERPRINTING BASED LOCALIZATION

2. RELATED WORK

We discuss prior works on indoor localization centred on Wi-Fi, or merger of Wi-Fi with other wireless or sensing technologies. Microsoft research lab introduced the idea of using Wi-Fi fingerprints for indoor localization in RADAR [16]. It used Wi-Fi fingerprints received at the Wi-Fi APs from the laptop carried by a user. Subsequently, they used both triangulation and kNN to estimate coordinates of user. They used empirically measured as well as theoretically computed Wi-Fi RSSI values for location estimation specifying 2-3m median error.

Amalgamation of Wi-Fi and BLE (Bluetooth Low Energy) FPs was suggested by Matthew Cooper et. al. [17] based on boosting technique for room level localization. They used pre-installed Wi-Fi APs and specifically installed BLE beacons FPs using a smart phone to train one classifier per room invoked in One-Vs-All notion. They used Decision Stumps with modified AdaBoost algorithm. They show cased its results on BLE only, Wi-Fi only and both BLE+ Wi-Fi.They used percentage accuracy and localization query response time as performance measures. They attained 94% accuracy of their per room classifiers with 0.0056 sec response time using Wi-Fi only approach and 96% accuracy with 0.0043 sec using hybrid Wi-Fi and BLE method. Their experimental setup included quite dense AP constellation of 159 BSSIDS covering approximately 1900 m^2, resulting in 0.009 APs visible per m^2.

A Wi-Fi based indoor localization was proposed by Nan Li et. al. [15], suggesting affinity propagation clustering with PSO (Particle Swarm Optimization) based ANN. PCA (Principle Component Analysis) was used on input data for dimension reduction before applying clustering on it. A separate ANN was trained for each natural FPs cluster found by affinity propagation clustering. At prediction stage, captured FP was passed through cluster matching to assign cluster label. Furthermore, the respective pre-trained ANN was used to estimate x,y coordinates. Their proposed approach produced mean error of 1.89 m and 90% error of 2.9 m.

Chunjing et. al. [14], proposed usage of Wi-Fi signals based on HNB (Hidden Naïve Bayes) classifier. They focused on finding best discriminating APs and elimination of redundant APs for each RP. They used modified Relief F with Pearson's correlation coefficient to eliminate these APs. Furthermore, they performed clustering on the filtered data. They determined similarity of RPs belonging to a specific cluster based on threshold of minimum size of common subset of best discriminating APs. Then one HNB is trained per cluster. To estimate location, cluster matching and respective HNB is invoked to estimate x, y coordinates.

Wan et. al. [18], suggested PSO based training of back propagation ANN using RSSI values of RFID tags. They performed data normalization in [0,1] range as pre-processing step along with Gauss filtering. Their estimated user location in terms of x,y coordinates resulted in mean error of 0.3448 m.

Sun et. al. [19], used RSSI samples, camera and room map information to optimize propagation model parameters. This optimization was done using trilateration and crowdsourced data at few locations. The crowdsourced RSSI values are used to correct trilateration localization errors. Furthermore, they used the panoramic camera and room map to detect human object on the observed image and find its pixel location. The pixel location on the image is mapped to the room map using ANN, reporting mean error of 3.75 m.

Wietzykowsk et. al. [20] used visual space identification algorithm FAB-MAP for indoor localization using Wi-Fi FPs, using UJIIndoorLoc. They presented performance results in x,y coordinates. They reported accuracy as correct prediction of both building ID and floor number combined. i.e. both were identified correctly. Their approach provided minimum error of 8.21 m between actual and predicted location.

RFID tags' RSSI fingerprints were incorporated by Calderoni et. al. [21], targeting room level accuracy. They first performed clustering on the dataset collected in an Italian hospital covering 4000 m^2 area, using k-means variant. After finding clusters in data, they trained RFID based classifiers per cluster for room label prediction. They estimated room labels from all those RF classifiers for which the observation's matching score is higher than a predefined threshold. The final room label estimation was calculated as maximum of the obtained confidence scores from these classifiers.

Above mentioned are some selected current works using RF signals for indoor localization, predominantly Wi-Fi or hybrid of Wi-Fi. Frameworks which are based on k-NN or its modifications such as [13,22-23] has the inherent problem that at run time for location prediction, template matching with the complete dataset is required. Henceforth, they face scalability issues in terms of increase in response time with larger or increasing datasets. Likewise, lately ANNs and deep learning has been frequently applied in indoor localization. ANNs require resource consuming training and can take long time for the training procedure to converge. But on the other hand, location prediction at run time needs merely minimal computation from pre-trained ANN. This generates almost trivial response time. Approaches such as [15,24-26] using ANNs has many hyper parameters to tune such as the suitable ANN architecture, learning function and number of neurons per layer, number of layers. ANN based approaches have shown considerable potential for indoor localization in terms of accuracy and response time, Monte Carlo hyper parameter tuning and testing is enormously resource intense regarding training duration and processing power. Majorly heuristics are employed for recommending an architecture based on ANNs.

In case of change in location of APs or addition/ elimination of new APs, fingerprinting based solutions need to be retrained on the restructured dataset. Repeated retraining puts ANN based approaches at a disadvantage. Some iterative methods such as Boosting cannot be parallelized to speed up training process. Some solutions suggest using classifiers in One-Vs-All notion [17], which demands that for each location estimation all the classifiers are invoked to generate the final output. In such approaches, as the size of buildings or area covered increases, the response time of the system also tend to increase.

We propose an easy to train multiclass classifier which is convenient to deploy regarding processing complexity, offers adequate necessary accuracy and delivers real-time response time. Hence, considering indoor localization as multiclass classification problem, we recommend it to be solved by one Ensemble classifier. This approach avoids scalability issues arose in case of One-Vs-All multiple classifiers producing similar accuracy. Along with that RF Ensemble needs only three tuneable hyper parameters namely number of trees, maximum number of splits and random number of predictors as the basis of split. Consequently, for rapid and frequent training and retraining required for real world practical deployments of indoor localization system, RF suits well.

3. PROPOSED FRAMEWORK

3.1 Localization as a Multiclass Classification Problem

We consider solving indoor localization as multiclass classification problem. Room level identification is required as an output symbolizing each room with a particular class label L_i. A smartphone with de facto wireless adapter card, can read Wi-Fi RSSI values from total M APs at a given location at a given time. As a result, every sample of reading a.k.a. fingerprint $FP_i = \{RSSI_{i1}, RSSI_{i2}, RSSI_{i3}, \ldots, RSSI_{iM}\}$ is generated. $RSSI_{ij}$ symbolizes the received signal strength value in dBm in ith FP sample from jth AP. These RSSI FPs are then pre-processed (data normalization and/or missing value treatment), marked with ground truth class label L_i and stored in database. Function F, learnt from previously collected FP database, is invoked to transform the observed FP to predict a specific class label/room ID as depicted in Equation (1).

$$L_x = F(FP_i) \tag{1}$$

The building area is split into square grid of N cells, each having dimensions AxB m. Centre of every grid cell is marked as (RP) M number of AP are visible in the building which may reside in the same building or in neighbouring buildings and are constantly transmitting wireless signals.

3.2 Training

On every RP, multiple readings are recorded, marked with ground truth label L_i for training of the localization engine. Collection of these FPs form the dataset based on which localization engine is trained. At run time, for location prediction, the pre-trained localization engine is used to estimate the class/room label based on acquired FPs from unknown location L_x as described by Fig. 2.

RDF Ensemble [27] was chosen from different classifiers to be used in localization engine. According to discussion in Section 2, the need for accuracy as well as real time response time for a localization framework can both be fulfilled by RDF Ensemble. RDF Ensemble provides accuracy and efficiency on huge databases, noise robustness, and can also handle missing values in data. As bootstrapping is used in RDF Ensemble, it provides reduced variance without increasing the bias as different subsets of training dataset with replacement ensure the decision trees are uncorrelated in nature. RDF Ensemble trained for this work needed an average of ~50 seconds with training repeated six times. Furthermore, trees are bagged in RDF enabling parallel training of various trees to further accelerate the training procedure. In prediction phase, the response time of Random Forest was trifling. These values were determined during our experiments and hardware details of experimental set up are discussed in Section 4. Along with that, it also generates the confidence level by each decision tree and can be used to calculate importance of predictors.

RF belongs to ensemble classification category in which tree predictors are combined. Let us suppose, TS denotes total number of samples in the training set, PN is the total number of predictors, the value specifying the number of input predictors to be used to split at a node of the tree is f where f <<PN and cNum is the total number of classes in all samples. RF uses Gini Index Equation (2) for finding best split. P_j represents the relative frequency of $Class_j$ in TS.

$$\mathrm{Gini}(TS) = 1 - \sum_{j=1}^{cNum} \left(P_j\right)^2 \tag{2}$$

Method presented in Table 1 is used to train RDF:

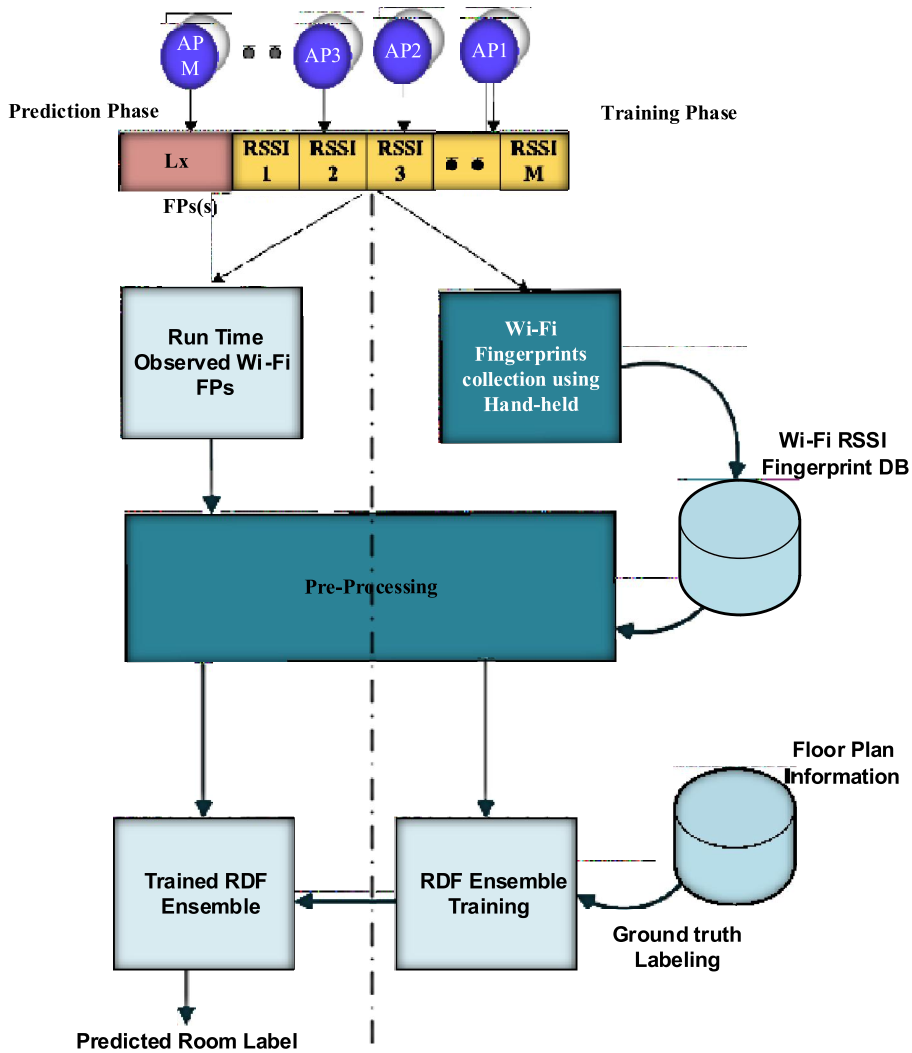

FIG. 2. PROPOSED LOCALIZATION FRAMEWORK

TABLE 1. RDF CLASSIFICATION METHOD

RDF Classification Method

Input: Training dataset having total PN predictors, Number of Trees TNum, Maximum Number of Splits MaxS, Random Number of Predictors f

Output: Predicted location L_x

Step-1: For i = 1 to TNum

(i) From the training, select a bootstrap sample set BS of size TS with replacement.

(ii) Produce a random forest Tree T_i to BS, by recursively iterating the following points for each terminal node of the tree, until the maximum number of splits MaxS is reached.

(a) Randomly pick f predictors from the PN predictors (f<<PN).

(b) Select the best predictor/split-point among the f using Gini Index Equation (2).

(c) Split the node into two child nodes.

Step-2: Output the ensemble of trees $\{T_i\}_1^{TNum}$.

At a new point x, prediction is computed as follows:

Assume $L_j(x)$ be the room prediction of the j^{th} random forest tree.

$L_{rf}^{TNum}(x)$ = majority vote $\{L_j(x)\}_1^{TNum}$

3.3 Prediction

Prediction Phase after training phase is very light in terms of computation time. For predicting the location at a specific point, the gathered RSSI FPs are averaged out and then the resulting FP is fed to every decision tree in the trained ensemble. A class label is predicted by each tree and the final class/room label is calculated by majority vote. Confidence for each class is also computed by RDF ensemble.

4. EXPERIMENTAL RESULTS AND ANALYSIS

This section discusses the hardware, software equipment utilized and experiments' particulars conducted to assess the performance of the proposed localization framework. The performance measures used for evaluation and comparison include accuracy percentage, precision, recall, F1-measure, miss rate/FNR (False Negative Rate), FDR (False Discovery Rate), training and response time.

System with Intel Xeon 64-bit processor X5650 clocked at 2.67GHz was used for experimentation having 24GB RAM, 64-bit Windows 10 Education. Real world dataset was collected at ground floor of Software Engineering Centre, University of Engineering & Technology, Lahore. The dimensions of the building are 39x31m^2 (approximately1209 m^2) which includes class rooms, labs, staff offices and open hallways. The exact and detailed building's floorplan is represented in Fig. 3 in which all room labels as well as RPs are marked. Open hallways (L12) surround a small lawn in the centre of the building. L1-L10 mark the closed rooms, L12 is the open connected hallway and two partially open spaces L11 and L13 surrounded by walls at 3 sides are specified as semi open rooms. This notion of rooms is utilized due to walls playing a significant role in fluctuation of Wi-Fi RSSI readings [28-29].

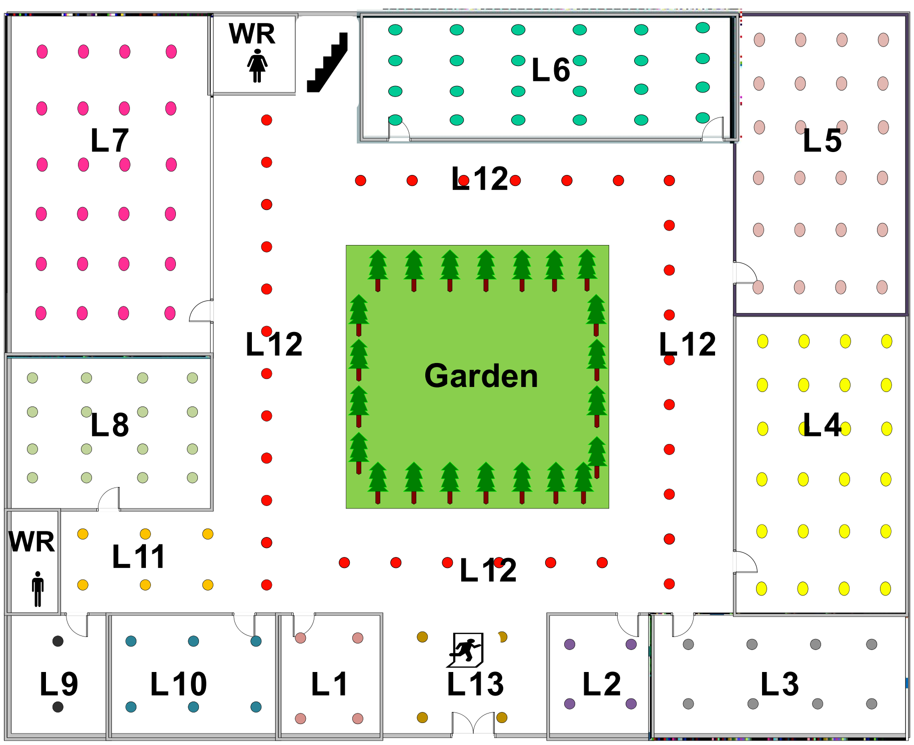

FIG. 3. ROOM LABELS AND MARKED RPS PER ROOM

The shape of SE centre is almost square. The total space was divided into square grid cells of 1.5x1.5 m. Every FP sample was collected at the centre point of each such cell labelled as RP. Fig. 3 pictorially represents these total 180 RPs marked in each room.

Total 54 APs were visible during the dataset collection. Three APs were installed in SE Centre and other 51 APs were located at known locations, in the other neighboring departments' buildings, which belong to university infrastructure. In experiment area of 1209 m^2, 3 APs were present, resulting in 0.002 APs per m^2. Following four steps constituted the course of experiments.

- **Dataset Collection:** We developed an Android app for data collection. Samsung Galaxy J5 (2016), hosting that Android App was used to scan all observable APs' readings at various RPs in 2.4 and 5 GHz bands. As identity of APs, their MAC addresses are used during dataset collection. A total of 20 samples was accumulated at each RP; 5 samples facing each direction N, E, S and W. As in all rooms of varying sizes, 180 RPs were marked for the experiments, 20 samples per RP produced 3600 samples in the dataset. Fig. 4 depicts the samples per room.

- **Dataset Pre-Processing:** RSSI values collected ranged from -50dBm (demonstrating high RSSI ~ close AP) to -95 dBm (low RSSI ~ far-off AP). On different RPs, some APs were visible and some were not, due to which several missing values are present in the dataset. In all dataset samples, these missing values were interchanged with -100 dBm. We did not find data normalization useful for the collected dataset which is further expressed in section 4.4. Henceforth, data was not normalized.

- **Training Phase:** RDF ensemble based localizer was trained on the pre-processed dataset samples labelled with room IDs. RDF ensemble has few tunable hyper parameters which include number of trees, fraction of predictors for each node decision and maximum allowed number of splits for controlling tree depth. Fig. 5 represents the relationship among these hyper parameters and obtained accuracy.

  After extensive experimentations 115 trees, 14 randomly chosen predictors, with 512 maximum allowed splits were determined to be optimal generating maximum accuracy, precision and recall.

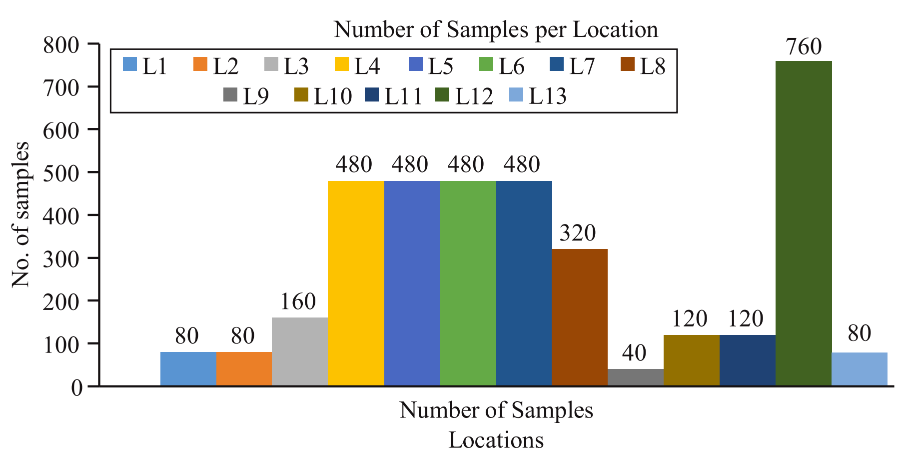

FIG. 4. NUMBER OF SAMPLES PER ROOM

- **Prediction Phase:** Trained RDF ensemble was installed on a machine for location estimation. Several RSSI FPs were collected, their mean was computed and was sent for run time location prediction. This machine can be set up as a server too, companion applications can contact it for location determination of subscribed users. The time duration from when a localization query was generated to the time when the estimate was calculated on average is termed as response time.

4.1 Performance Evaluation Measures

The performance comparison and evaluation of the proposed framework is presented in Table 2, a comparison is drawn with well-known recurrently utilized k-NN based solution averaged over 16 configurations using Euclidean distance and FFMLANN (Feedforward Multi-LayerArtificial Neural Network). FFMLANN based localization solution was averaged over 4-layer, 3-layer and 2-layer over 4, 3, and 3 configurations respectively with SCG (Scaled Conjugate Gradient) and RBP (Resilient Back Propagation) both. The best performance shown by all evaluation parameters, are emphasized in Table 2 in boldface.

The Equations (3-8) mention the formulae to calculate performance parameters:

$$\text{Accuracy}\left(\%\right) = \frac{\text{Tp} + \text{Tn}}{\text{Tp} + \text{Tn} + \text{Ep} + \text{Fn}} \tag{3}$$

$$\text{Precision} = \frac{\text{Tp}}{\text{Tp} + \text{Fp}} \tag{4}$$

$$\text{Recall} = \frac{\text{Tp}}{\text{Tp} + \text{Fn}} \tag{5}$$

$$\text{PF1 - Measure} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

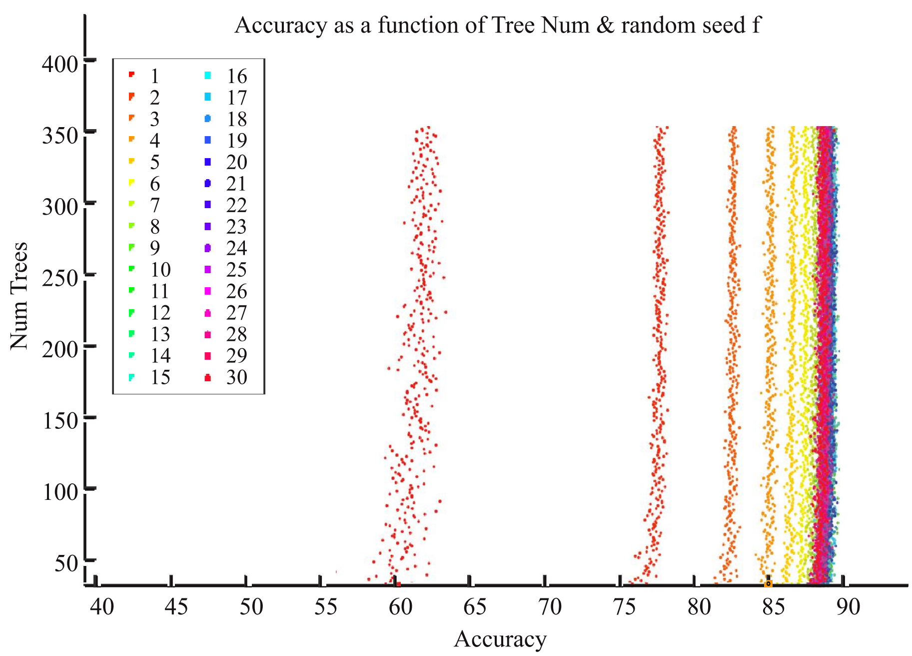

FIG. 5. RELATIONSHIP AMONG HYPER PARAMETERS AND ACCURACY

Miss Rate = FNR = 1- Recall (7)

False Discovery Rate = FDR = 1 – Precision (8)

Section 3.1 previously discussed our approach to solve indoor localization as a multiclass classification problem, the evaluation of parameters listed in Equations (3-8) was performed in accordance with it. Fig. 3 describes that total 13 location labels were marked L1-L13 in the experimental set up of SE Centre building of our university. Hence the predicted labels by the proposed approach along with the ground truth labels were used to generate a confusion matrix for all 13 location labels including information related to TP, TN, FP and FP. These entities were further used for the evaluation parameters listed above. All the values specified from Table. 2 onwards for accuracy, precision, recall, F-1 measure, FNR and FDR have been averaged over all 13 labels/classes depicting the overall performance of the system for the whole building covered.

Our proposed framework leaves behind kNN and ANN based localization solutions regarding accuracy, precision, recall, F1-measure, FNR and FDR which is evident from Table 2. kNN does not require any training at all, rather it simply involves storing pre-processed samples along with marked labels. Related work section earlier discussed suitability of ANN in terms of response time, it is also evident from Table 2 that ANN's response time results are superior than both kNN and proposed framework but the time difference is trifling (on scale of 10E-01 seconds). It should be noted that the training time for all approaches presented in Table 2 was calculated without parallel processing. Training time of RDF ensemble can be reduced by employing parallel computation which is not possible in case of Boosting based methods and few training algorithms of ANNs. kNN and RDF ensemble's response times are similar, approximately 0.0002 seconds needed to calculate one location estimation. Therefore,

TABLE 2. PERFORMANCE EVALUATION AND COMPARISON OF PROPOSED FRAMEWORK

	Accuracy (%)	Precision (%)	Recall (%)	F1 Measure	FNR	FDR	Training Time for 10 Folds	Avg. Training Time for 1 Folds	Response Time DS	Response Time Per Sample
kNN	82	75	70	0.71	0.30	0.25	-	-	0.69	1.92E-04
4-Layer FFMLNN (SCG)	84	82	75	0.77	0.25	0.21	103.13	10.31	0.04	1.03E-05
4-Layer FFMLNN (RBP)	85	85	81	0.82	0.19	0.16	59.01	5.90	0.03	8.46E-06
3-Layer FFMLNN (SCG)	85	84	80	0.82	0.20	0.16	92.01	9.20	0.03	9.45E-06
3-Layer FFMLNN (RBP)	85	83	79	0.81	0.21	0.16	45.19	4.52	0.02	6.9E-06
2-Layer FFMLNN (SCG)	71	77	73	0.74	0.26	0.23	92.96	9.30	0.05	1.42E-05
2-Layer FFMLNN (RBP)	63	84	80	0.82	0.20	0.16	60.02	6.00	0.05	1.36E-05
RDF Ensemble	89	89	83	0.85	0.17	0.11	59.6	5.96	0.76	2.12E-04

making RDF ensemble an appropriate candidate for localization with enhanced accuracy, few tunable parameters requirement in training process, adequate training time and real time response time. Matthew et. al. [17] obtained 94% accuracy for room level prediction with response time of 0.0056 sec using Wi-Fi only approach and 96% with 0.0043 sec incorporating both Wi-Fi and BLE signals. However, it must be noted that there was dense custom deployed population of APs having 159 BSSIDS covering 1900 m^2, which translates into 0.009 APs visible per m^2. Whereas in our experimental set up it was 0.002 APs per m^2. In [21], 86% accuracy in a hospital environment was achieved using Random Forest for room level prediction. Yasmine and Pei [30], used normalized rank based SVM (Support Vector Machines) for Wi-Fi based room level prediction, their presented results expressed 98.75% accuracy in 93.75% of the tested cases and 100% accuracy in 56.25% of cases, but the generic overall building level accuracy details were not given. Experimental area covered was described in terms of 88 shops in a mall with 185 APs but this information is insufficient to calculate number of APs visible per m^2.

It should be noted that all 3 approaches specified in Table 2 have diverse tunable hyper parameters. In case of kNN based localization solutions, there are mainly three hyper parameters namely k (which determines the number of neighbors considered to take part in majority voting), similarity measure (distance measure used to compute pairwise similarity for the whole dataset) and neighbor weight (if all the k neighboring instances/samples are given equal weight or different weights during final decision estimation based on majority weight). For RDF Ensemble, details have already been provided in Table 1 presented in Section 3.2. For ANN centered approaches, the list of hyper parameters is far greater than the rest of the two approaches mentioned in Table 2. The list of hyper parameters for ANN includes type of the neural network (configuration of neurons' connections), number of hidden layers, number of neurons in each hidden layer, the learning algorithm used during training phase, training performance evaluation function, regularization parameter used if any etc. The hyper parameters list and possible assignments also vary slightly for different types of neural networks. This information is summarized in Table 3.

TABLE 3. HYPER PARAMETERS REQUIRED BY ALL THREE APPROACHES

	List of Required Hyper Parameters	Sample Hyper Parameter
kNN	k	1,2,7
	Distance measure	Euclidean, Dilca, Manhattan
	Neighbour weight	Nil, 1/distance, 1-distance
	Network type	Feedforward, Recurrent, SOM
	Number of hidden layers	2, 4, 70
FFMLNN	Number of neurons per hidden layer	100-300, 20-20-500-500
	Learning algorithm	SCG, RBP
	Training performance function	Mean squared error, Cross entropy
	Regularization parameter	0.03, 0.5
RDF Ensemble	Number of trees	13, 600
	Number of random features f	3, 17
	Number of Max Splits(Tree depth control)	32, 512, 1024

In terms of hyper parameter tuning, kNN and RDF Ensemble based approaches have maximum 3 parameters which makes finding the suitable configuration for any given problem easier than their ANN based counterpart. ANNs have shown promising results for indoor localization but usually the proposed solution is based on heuristics rather than exhaustive Monte Carlo or detailed experimentation. It is also interesting to note that during experimentation, it was observed that at times a very slight change in number of neurons in a hidden layer(s) or change of training algorithm modifies the overall performance of the system drastically which is not found to be linearly or inversely related to number of hidden layers and/or number of neurons per hidden layer(s). Also the convergence during training is highly variant and does not show any specific trend. From Table 2, it can be observed that response time of 4-layer and 3-layer FFMLN is smaller than 2-Layer FFMLN and training/ convergence time for both SCG 2-Layer and 3-layer FFMLN was ~92 seconds whereas RBP 3-layer FFMLN took ~45 seconds to converge. Also 2-layer FFMLN response time was greater than 3-layer and 4-layer configurations. Whereas as evident from sections 4.5-4.7, for RDF Ensemble the relationship between increasing all 3 hyper parameters and training-testing time is straightforward i.e. linearly increasing with growing number of trees, f and max splits allowed. kNN based approaches also give the researcher an estimate of the training-testing time with increasing k, distance measure and weight assignment to neighbors. With increasing k, complex distance measure calculation and weight assignment calculation of neighbors, the training-testing time will also increase. Major drawback of kNN is that for a single location prediction, similarity is measured with all the instances in the dataset which will increase the response time with growing dataset size.

Further experiments were conducted to investigate impact of:

- Increase in Reference Points
- Increase in Access Points
- Performance Measures with data normalization
- Increase in tree numbers
- Increase in Max Splits allowed
- Increase in Random seed f
- Complexity Analysis of Proposed Framework

4.2 Increase in Reference Points

We observed during preliminary data collection and experimentation that majority of the misclassifications were present in open areas such as open hallways in our experiment area. We doubled the RPs in the hallways to investigate the impact of RP density. These are labelled as L11, L12 and L13 in Fig 3, where we kept the RPs in remaining rooms the same resulting in total 4524 FPs collection. Fig. 6 sheds light on the outcomes calculated with the optimal configuration found.

This experiment shows that increase in number of RPs can increase prediction accuracy to a certain point after which increasing RPs does not improve performance, rather it results in performance degradation. This occurs because of amplified FP resemblance between RPs which are very close to each other, making it tough to discriminate.

4.3 Increase in Access Points

We determined more distinguishing APs in order to find impact of AP density on the framework's performance. We increased number of APs ranging from 1-10 in increments of 1, after 10 in increments of 5 and all performance measures were computed for each set which is described in Fig. 7. The accuracy remained under 80% up to five APs. After crossing 5 APs, largely the performance kept improving suggesting more number of APs can aid increase the system's ability to differentiate between locations.

4.4 Performance Measures with Data Normalization

We tested the optimal configuration without normalizing data as well as with normalized data. For data normalization both schemes, mapping raw RSSI values to [-1,1] and [0,1] ranges respectively were evaluated. The resulting performance measures are abridged in Table 4. Henceforth we chose not to incorporate normalized data representation.

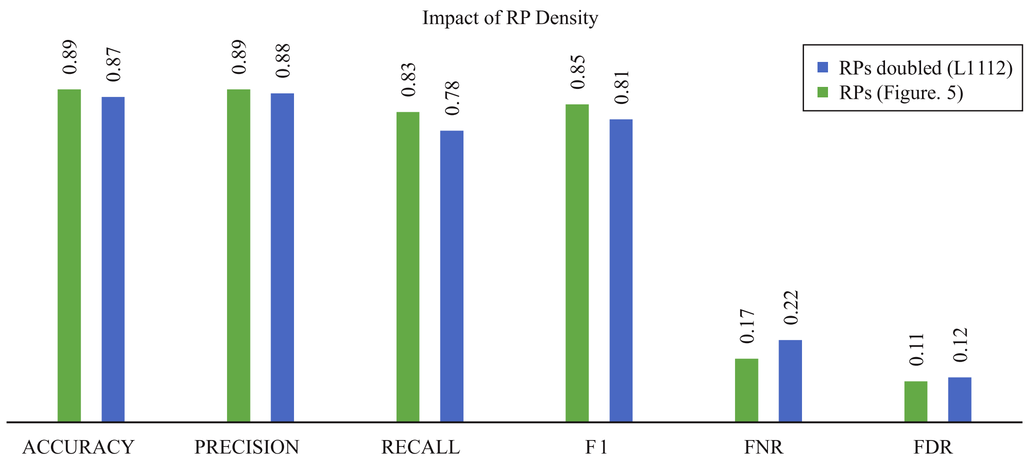

FIG. 6. IMPACT OF RP DENSITY

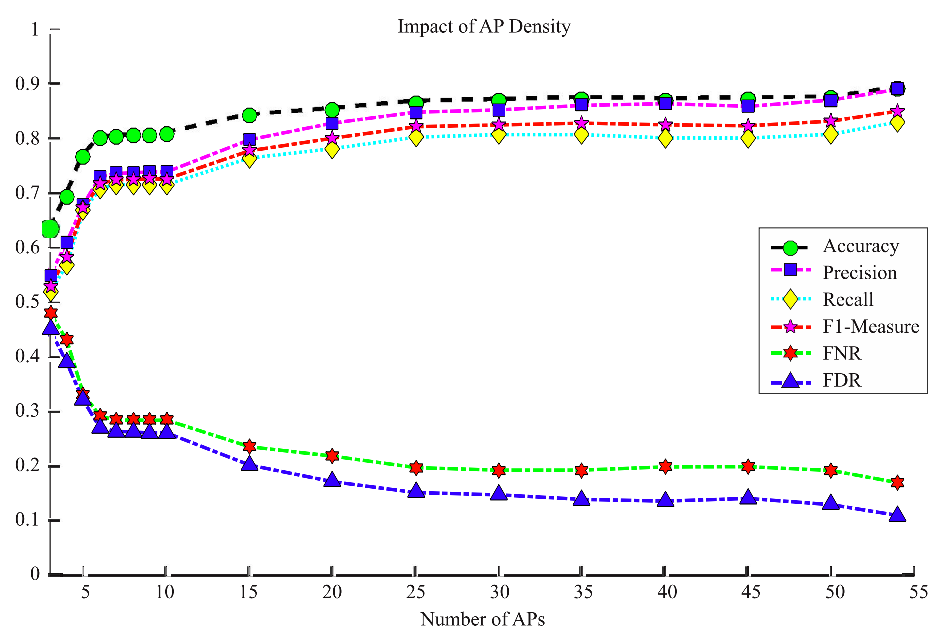

FIG. 7. IMPACT OF AP DENSITY

4.5 Increase in Tree Numbers

RDF Ensemble's effectiveness and efficiency are highly affected by number of trees included in the ensemble. To capture the impact of growing more trees, we varied them from one to a few hundreds. The initial count of one tree is used as the base for comparison with performance attained with greater number of trees. Figs. 8-9 describe relationship among number of trees and all performance measures. It can be observed from Fig 8 that 60+ number of trees provided major improvement regarding performance measures, the optimal performance reached at 115 and more increase in number of trees did not impact much.

Training time expressed nearly linear relationship with number of trees included in ensemble. However, prediction time of RDF Ensemble remained constant for the entire range 1-400 with total difference of mere 1.5E-03 seconds.

TABLE 4. OPTIMAL CONFIGURATION WITH AND WITHOUT NORMALIZED DATA

Data Format	Accuracy	Precision	Recall	F1	FNR	FDR
Not Normalized	0.89	0.89	0.83	0.85	0.17	0.11
[-1,1]	0.84	0.82	0.74	0.77	0.25	0.17
[0,1]	0.85	0.83	0.75	0.78	0.24	0.16

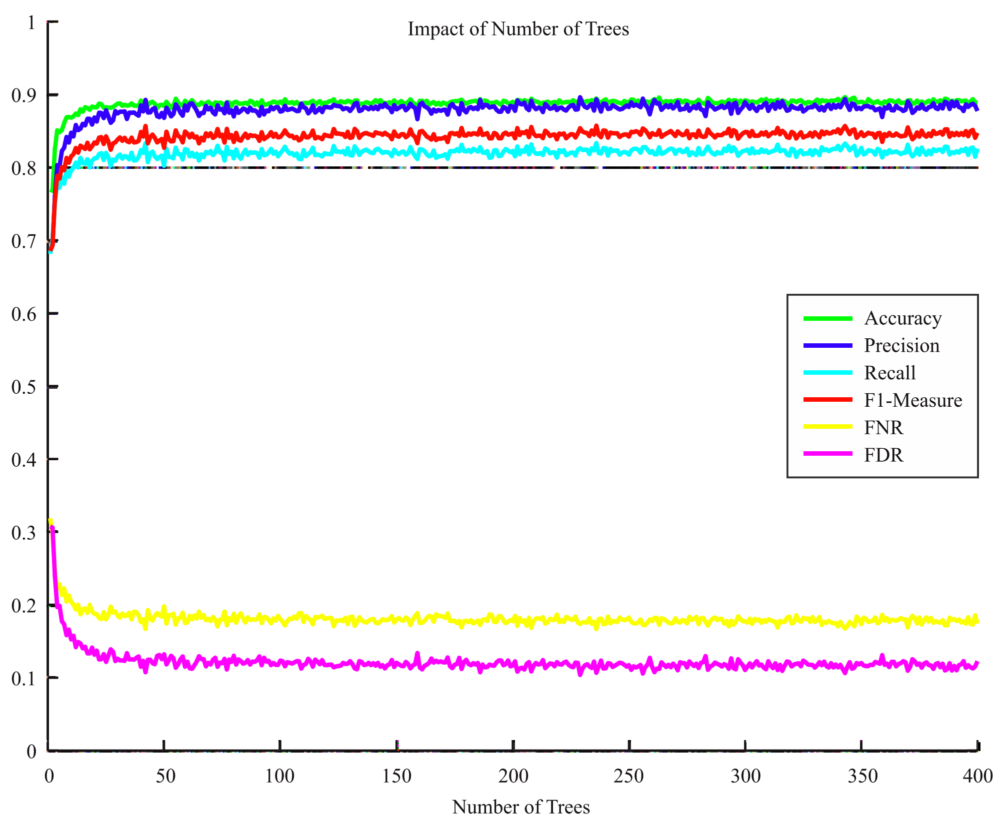

FIG. 8. IMPACT OF NUMBER OF TREES

4.6 Increase in Maximum Splits Allowed

We established impact of regulating maximum number of splits for trees growth. The variance among trees in ensemble can be controlled by maximum number of splits allowed per tree. It influences effectiveness of the framework regarding overfitting avoidance besides both training and testing time consumption. We varied maximum number of splits over the range 1-2048 in powers of 2 based on the dataset size. The consequential performance measures, training and testing time impact is presented in Fig. 10 (splits 64-2048) and 11 respectively.

The finest blend of accuracy, precision, recall, F1-measure, FNR and FDR is obtained at maximum number of splits kept at 512 which is clearly observable from Fig. 10. The second best performance is reached at 1024 maximum allowed splits for controlling tree depth in ensemble.

Among training and testing time, with changing maximum splits, growth is observed as shown in Fig. 11. Overall training time increased quickly but after 256 maximum splits it remained almost stable at 6 seconds. The described outcomes are mean training time per fold and mean response time per sample.

4.7 Increase in Random Seed f

Random seed f, was expressed in Table 1 as the number of predictors selected randomly for splitting decision tree. This is a distinctive feature of RDF ensemble because it is useful in adjusting bias concerning any particular feature in the course of training. We assessed the

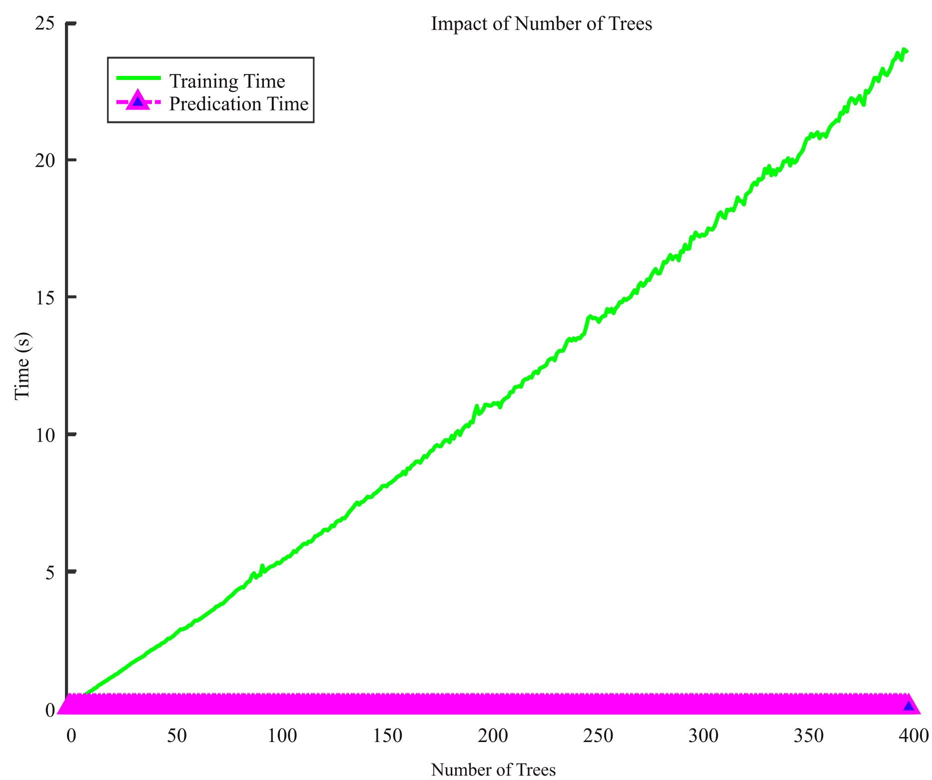

FIG. 9. IMPACT OF NUMBER OF TREES ON TRAINING AND TESTING TIME

performance by increasing f. Fig. 12 exhibits the deviations in all evaluation parameters except time which are characterized discretely in Fig. 13. Time was found to be growing linearly. Nevertheless, the overall time variation for testing for all f features was a trifling 1.3E-04 seconds.

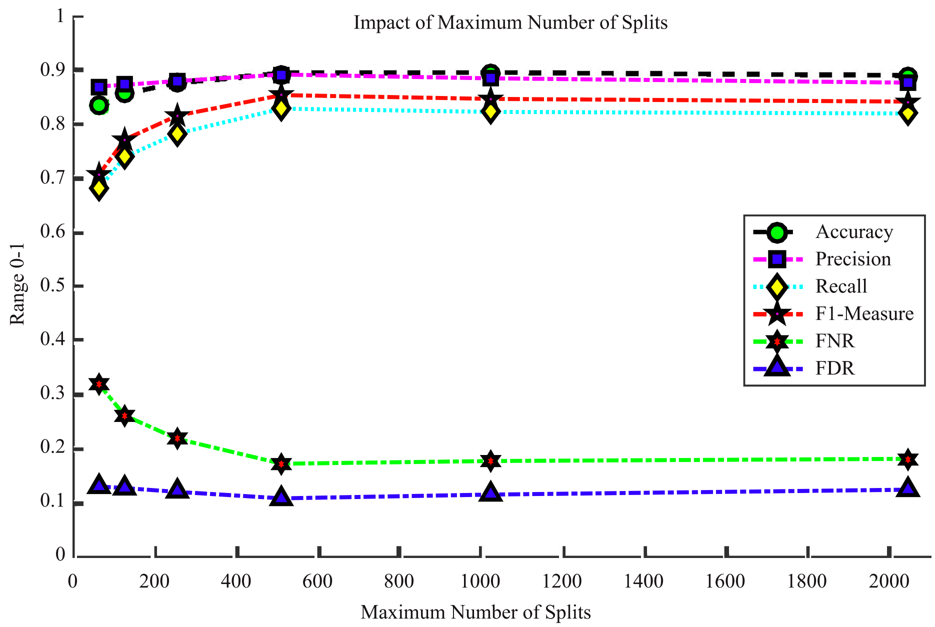

FIG. 10. IMPACT OF NUMBER OF SPLITS

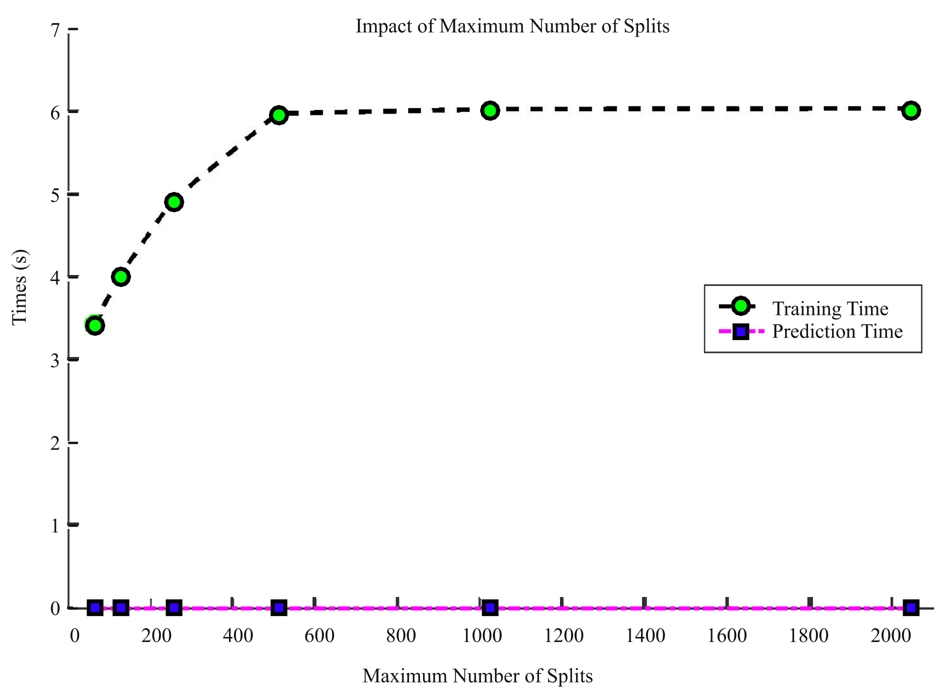

FIG. 11. IMPACT OF NUMBER OF SPLITS

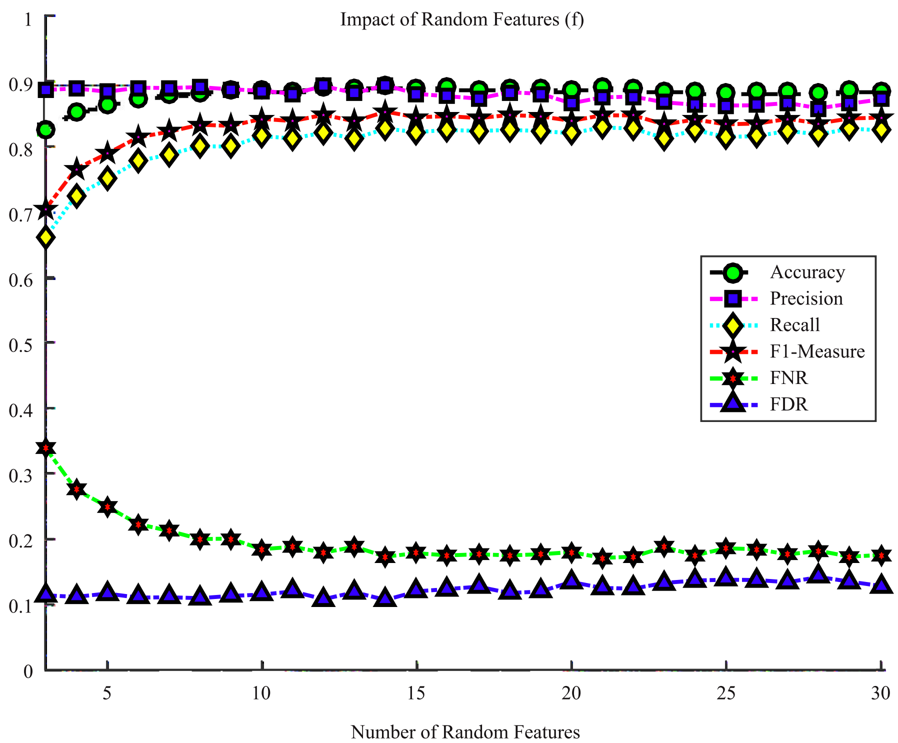

FIG. 12. IMPACT OF RANDOM SEED SELECTION

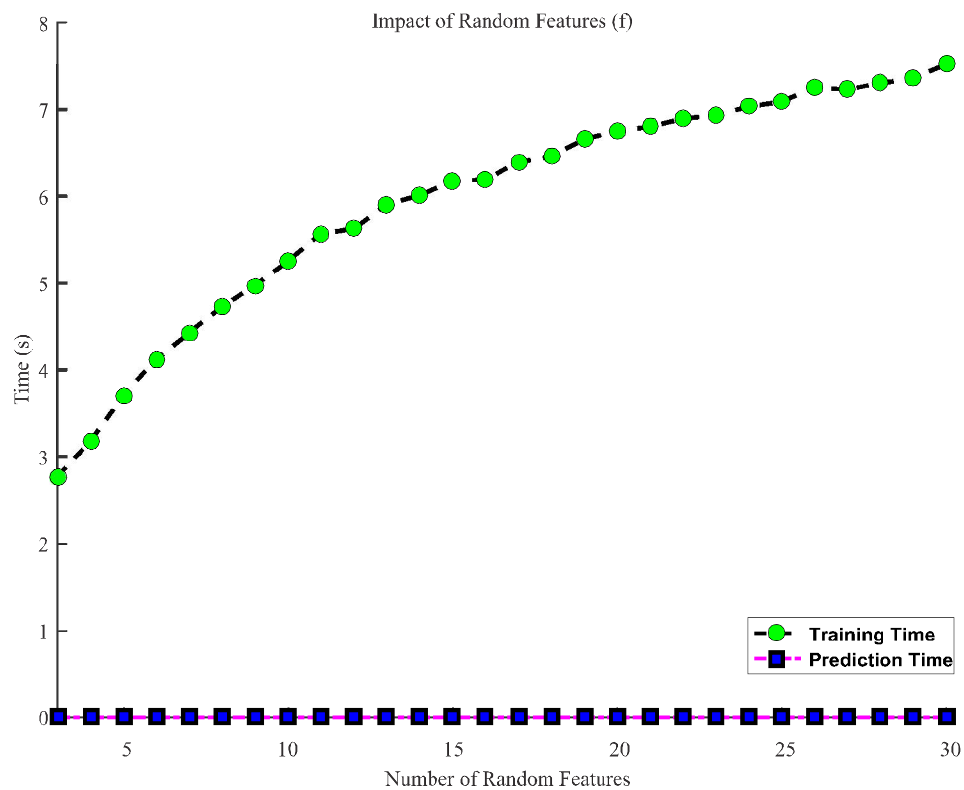

FIG. 13. IMPACT OF RANDOM SEED SELECTION ON TRAINING AND TESTING TIME

4.8 Complexity Analysis of Proposed Framework

Complexity of the proposed framework can be evaluated by the memory and training-testing time consumed. Section 4.2-4.7 shed light on the performance measures as well as training-testing time behavior impacted by all three hyper parameters and AP-RP density. Apart from hyper parameters' impact on the proposed framework, dataset size is another important aspect which can influence the performance of the system, training-testing time and memory requirements. The following experiments were performed specifically to examine the complexity of the RDF Ensemble. The first fold consisted on increasing number of APs/predictors (columns in dataset) to grow the dataset size. In second fold, the number of APs were kept constant (i.e. all APs = 54), but the number of rows in the dataset were increased by increasing the RPs in the open and semi open areas of the building. By increasing number of APs, the number of predictors/features in the dataset increases. We used the optimal configuration as previously described in Section 4 to obtain the results for all data configurations with APs increased from 1-54. The results are useful to determine the general trend of the system with increasing APs/predictors (larger building and/or more than one building) in terms of the training time required, response time, memory requirements for storing trained RDF ensemble, a single input RSSI value and output room/class label. Table 5 presents the summarized results of the performance measures and Table 6 shows the training time, response time and memory requirement.

TABLE 5. PERFORMANCE MEASURES WITH INCREASING APS

Number of APs	Accuracy	Precision	Recall	F1	FNR	FDR
1	0.43	NaN	0.31	NaN	0.69	NaN
2	0.55	NaN	0.41	NaN	0.59	NaN
3	0.64	0.55	0.52	0.53	0.48	0.45
4	0.69	0.61	0.57	0.58	0.43	0.39
5	0.77	0.68	0.67	0.67	0.33	0.32
6	0.80	0.73	0.71	0.72	0.29	0.27
7	0.80	0.74	0.72	0.72	0.28	0.26
8	0.80	0.74	0.72	0.73	0.28	0.26
9	0.80	0.74	0.72	0.73	0.28	0.26
10	0.81	0.74	0.72	0.73	0.28	0.26
15	0.84	0.80	0.76	0.78	0.24	0.20
20	0.85	0.83	0.78	0.80	0.22	0.17
25	0.86	0.85	0.80	0.82	0.20	0.15
30	0.86	0.85	0.81	0.82	0.19	0.15
35	0.87	0.86	0.81	0.83	0.19	0.14
40	0.87	0.86	0.80	0.83	0.20	0.14
45	0.87	0.86	0.80	0.82	0.20	0.14
50	0.88	0.87	0.81	0.83	0.19	0.13
54	0.89	0.89	0.84	0.85	0.15	0.11

In Table 5, NaN is obtained if entire column in the confusion matrix for multiclass classification becomes 0 which indicates that all samples belonging to one particular class were misclassified. It can be deduced that with increasing number of APs, the system's capability to correctly distinguish different rooms also grew, major jump in accuracy was observed when number of APs became 4,5 and 6. Afterwards the increase caused slow and gradual performance improvement. Table 6 shows that as the number of APs/predictors kept increasing, the training time was proportionally increasing, the response time was also increasing but the overall difference over the entire range was on scale of 1.45E-04 seconds.

In terms of memory requirements, the space required to store the trained RDF Ensemble grew with increase in number of predictors but the difference of memory required for RDF ensemble trained on 54 APs (138.8 MB) and same trained on 1 APs (5.94 MB) is of 139386047 bytes ~ 132.92 MBs which indicates that this RDF ensemble is also suitable to be deployed on Android of iOS based smart phone for offline location prediction. Memory required for storing a single RSSI vector for 1-54 APs varied from 8-432 bytes. The memory required to store the predicted location label remained constant i.e. 8 bytes.

TABLE 6. COMPLEXITY ANALYSIS WITH INCREASING APS

Number of APs	Training Time (sec)	Testing Time 1 Sample (sec)	Memory Required RDF Ensemble (Bytes)	Memory Required 1 RSSI Vector (Bytes)	Memory Required Output 1 Sample (Bytes)
1	21.53	8.99E-05	6237991	8	8
2	35.64	1.30E-04	98479506	16	8
3	50.33	1.54E-04	144157405	24	8
4	54.61	1.76E-04	149291944	32	8
5	56.83	1.72E-04	152137363	40	8
6	55.40	2.14E-04	156898654	48	8
7	57.00	2.10E-04	157716393	56	8
8	58.77	2.15E-04	158420100	64	8
9	60.62	2.13E-04	158773955	72	8
10	62.32	2.48E-04	159325198	80	8
15	64.22	2.13E-04	148519985	120	8
20	64.68	2.49E-04	147208720	160	8
25	66.11	2.48E-04	147658675	200	8
30	65.02	2.37E-04	149743266	240	8
35	65.54	2.38E-04	149951289	280	8
40	64.05	2.35E-04	149053396	320	8
45	63.32	2.34E-04	147727991	360	8
50	62.81	2.27E-04	145723158	400	8
54	62.59	2.35E-04	145624038	432	8

The second fold of complexity analysis was performed by increasing number of samples in the dataset (increased number of rows) for which we used the dataset collected for experiments described in section 4.2. Table 7 compares the performance measures for original dataset (54 APs) with higher number of samples in open and semi-open spaces. It shows that by doubling the RPs in open and semi-open spaces the overall performance was degraded due to interference of signals.

This doubled RP dataset was used to study the effect of increased number of rows in dataset detailed in **Table 8**. The training time became 69.39 seconds instead of 62.59 but the testing time/response time remained almost same. Memory consumed for storage of RDF Ensemble increased by 1.57 MBs but the memory requirements for storing input RSSI vector and output room label remained exactly the same.

It can be deduced from the experimental results that both increasing number of APs and the total number of samples in the dataset affect the training-testing time as well as the memory required for storage of RDF ensemble and input RSSI vector. The number of APs is the dominant aspect as it increased all memory and time requirements except output label storage which is dependent upon room label encoding. Whereas, increased number of rows/ samples in dataset also affect overall performance, training-testing time, memory required for storing RDF Ensemble but it does not change the memory required for RSSI vector storage which is solely dependent upon the number of APs considered in the system.

5. CONCLUSION

Location can render meaningful context in ubiquitous computing and IoT. We believe improved indoor localization will broaden new horizons for IoT applications. Our proposed approach was validated through detailed experimentation. It provides improved accuracy, precision and real-time response time. Furthermore, it requires only three parameters to be tuned. Therefore, these characteristics make it feasible as fingerprinting based localization framework demanding frequent data recollection and classifier training. It achieved 89% accuracy for predicting room label for indoor localization in comparison with ANN (85%) and kNN (82%) based approaches.

TABLE 7. PERFORMANCE MEASURES WITH INCREASING RSSI SAMPLES

Number of Samples	Accuracy	Precision	Recall	F1	FNR	FDR
Original	0.89	0.89	0.84	0.85	0.15	0.11
Doubled in Open and semi-open spaces	87	0.88	0.78	0.81	0.22	0.12

TABLE 8. COMPLEXITY ANALYSIS WITH INCREASING RSSI SAMPLES

Number of Samples	Training Time (sec)	Testing Time 1 Sample (sec)	Memory Required RDF Ensemble (Bytes)	Memory Required 1 RSSI Vector (Bytes)	Memory Required Output 1 Sample (Bytes)
Original	62.59	2.35E-04	145624038	432	8
Doubled in Open and semi-open spaces	69.39	2.26E-04	147271534	432	8

An important real world deployment aspect to consider is the scalability of proposed framework which is directly dependent upon dataset size. This affects both the tuneable hyper parameter selection, effectiveness and efficiency in terms of training-testing time as well as required memory. Dataset size is determined by the total area covered by a single building and/or multiple buildings, number of visible APs, RP density and number of samples collected per RP. High number of samples per RP have been found during experiments for better location prediction capability but it only affects the training time resources in terms of memory and training time consumed. It does not change the response time or the memory required for input storage during prediction phase. Considering the total area, if a single building area is huge, with lesser APs the system performance will degrade as many small regions might be left uncovered by Wi-Fi signals. Sufficiently high number of APs are required to cover a huge building(s) resulting in larger number of predictors in the dataset whose impact is both on effectiveness and efficiency of the system. As previously discussed in section 4.3 and 4.8, increasing number of APs also helps in improved discrimination of different locations by the proposed system. However, more number of predictors translate into possibly more number of trees, higher f and/or max splits allowed resulting in higher memory and time consumption during training phase. Additionally, large number of APs/predictors means a large vector of RSSI values to be processed for prediction at run time incurring higher response time and storage memory for storage of large RSSI vector(s) and RDF ensemble. To solve these issues RDF ensemble can again be useful in the reduction of APs list/number of predictors by estimating the more important(distinguishing) AP, hence eliminating lesser important APs in the dataset. This additional processing step can reduce one dimension of the dataset in terms of predictors/features producing a reduced dataset. This reduced dataset can be used to train the proposed system requiring lesser memory and training time as well as reduced response time and run time memory required at prediction stage. In case of multiple buildings covering a vast area, merger of Wi-Fi and GPS based approach can be used; GPS to zoom onto a specific building (as GPS can provide building level resolution) and then invoking the Wi-Fi based RDF Ensemble for room level prediction. Hence, overcoming the aforementioned possible shortcomings by incorporating hybrid signals approach (GPS + Wi-Fi).

These points also determine our future direction where we plan to increase our dataset size encompassing multiple buildings and floors with varying environments. This will evaluate its performance in more detail and will test its scalability quantitatively. Various safety, security and evacuation guide applications can be built based on it for users in offices, universities and retail. Fingerprinting based localization demands frequent data collection which is both laborious as well as time consuming. We aim to incorporate crowdsourcing for fingerprints collection. Moreover, hybrid techniques such as merger with GPS, Bluetooth and PDR on basis of signals availability at any given time, are also possible to further enhance accuracy.

ACKNOWLEDGEMENT

The authors acknowledge Dr. Usman Ghani Khan, and Dr. Hafiz Shahzad Asif, Department of Computer Science & Engineering, University of Engineering & Technology, Lahore, Pakistan, for their technical and administrative support.

REFERENCES

[1] Soltani, M.M., Motamedi, A., and Hammad, A., "Enhancing Cluster-Based RFID Tag Localization Using Artificial Neural Networks and Virtual Reference Tags", Automation in Construction, Volume. 54, pp. 93-105, Elsevier, 2015.

[2] Rodrigues, M.L., Vieira, L.F.M., and Campos, M.F.M., "Fingerprinting-Based Radio Localization in Indoor Environments Using Multiple Wireless Technologies", IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 1203-1207, Canada, 2011.

[3] Luo, J., and Gao, H., "Deep Belief Networks for Fingerprinting Indoor Localization Using Ultrawideband Technology", International Journal of Distributed Sensor Networks, pp. 5840916, SAGE, 2016.

[4] Azizyan, M., Constandache, I., and Choudhury, R., "SurroundSense: Mobile Phone Localization via Ambience Fingerprinting", Proceedings of ACM 15th Annual International Conference on Mobile Computing and Networking, pp. 261-272, September 2009.

[5] Nagpal, P.S., and Rashidzadeh, R., "Indoor Positioning Using Magnetic Compass and Accelerometer of Smartphones", IEEE International Conference on Selected Topics in Mobile and Wireless Networking, pp. 140-145, August ,2013.

[6] Saeedi, S., Paull, L., Trentini, M., and Li, H., "Neural Network-Based Multiple Robot Simultaneous Localization and Mapping", IEEE Transactions on Neural Networks, Volume 22, No. 12, pp. 2376-2387, 2011.

[7] Pei, L., Chen, R., Liu, J., Chen, W., Kuusniemi, H., Tenhunen, T., Kröger, T., Chen, Y., Leppäkoski, H., and Takala, J., "Motion Recognition Assisted Indoor Wireless Navigation on a Mobile Phone", Proceedings of 23rd International Technical Meeting of the Satellite Division of the Institute of Navigation, pp. 3366-3375, Portland, September, 2010.

[8] Menke, J., and Zakhor, A., "Multi-Modal Indoor Positioning of Mobile Devices", Proceedings of International Conference on Indoor Positioning and Indoor Navigation, pp. 13-16, Banff, Canada, October, 2015.

[9] Ruotsalainen, L., Kuusniemi, H., and Chen, R., "Visual-Aided Two-Dimensional Pedestrian Indoor Navigation with a Smartphone", Journal of Global Positioning Systems, Volume 10, No. 1, pp.11-18, Springer Open, 2011.

[10] Khanzada, T.J.S., Memon, S., and Hashmani, A.A., "A Novel Method to Implement the Matrix Pencil Super Resolution Algorithm for Indoor Positioning", Mehran University Research Journal of Engineering & Technology, Volume, 30, No. 4, pp. 625-634, Jamshoro, Pakistan, October, 2011.

[11] Khanzada, T.J.S.,Ali, R.A., and Omar, A.S., "Time Difference of Arrival Estimation using Super Resolution Algorithms to Minimize Distance Measurement Error for Indoor Positioning Systems", Proceedings of IEEE Multi-Topic Conference, Karachi, Pakistan, 23-24 December, 2008.

[12] Khanzada, T.J.S., "Wireless Communication Techniques for Indoor Positioning and Tracking Applications", Ph.D. Dissertation, Faculty Electectrical Engineering & Information Technology, Magdeburg University, Magdeburg, Germany, 2010.

[13] Oussalah, M., Alakhras, M., and Hussein, M.I., "Multivariable Fuzzy Inference System for Fingerprinting Indoor Localization", Fuzzy Sets and Systems, Volume 269, pp.65-89, Elsvier, 2015.

[14] Song, C., Wang, J., and Yuan, G., "Hidden Naive Bayes Indoor Fingerprinting Localization Based on Best-Discriminating AP Selection", ISPRS International Journal of Geo-Information, Volume 5, No. 10, pp. 189, 2016.

[15] Li, N., Chen, J., Yuan, Y., Tian, X., Han, Y., and Xia, M., "A Wi-Fi Indoor Localization Strategy Using Particle Swarm Optimization Based Artificial Neural Networks", International Journal of Distributed Sensor Networks, Volume 12, No.3, pp. 4583147, SAGE, 2016.

[16] Bahl, P., and Padmanabhan, V.N., "RADAR: An In-Building RF-Based User Location and Tracking System", Proceedings of IEEE 19th Annual Joint Conference on Computer and Communications Societies, Volume 2, pp. 775-784, 2000.

[17] Cooper, M., Biehl, J., Filby, G., and Kratz, S., "LoCo: Boosting for Indoor Location Classification Combining Wi-Fi and BLE", Personal and Ubiquitous Computing, Volume 20, No.1, pp. 83-96, Springer, 2016.

[18] Wang, C., Wu, F., Shi, Z., and Zhang, D., "Indoor Positioning Technique by Combining RFID and Particle Swarm Optimization-Based Back Propagation Neural Network", Optik-International Journal for Light and Electron Optics, Volume 127, No. 17, pp. 6839-6849, Elsevier, 2016.

[19] Sun, Y., Meng, W., Li, C., Zhao, N., Zhao, K., and Zhang, N., "Human Localization Using Multi-Source Heterogeneous Data in Indoor Environments", IEEE Access, Volume 5, pp. 812-822, 2017.

[20] Wietrzykowski, J., Nowicki, M., and Skrzypczyñski, P., "Adopting the FAB-MAP Algorithm for Indoor Localization with WiFi Fingerprints", Proceedings of International Conference Automation, pp. 585-594, Springer, Singapore, March, 2017.

[21] Calderoni, L., Ferrara, M., Franco, A., and Maio, D., "Indoor Localization in a Hospital Environment Using Random Forest Classifiers", Expert Systems with Applications, Volume 42, No. 1, pp. 125-134, Elsevier, 2015.

[22] Zhang, M., Shen, W., and Zhu, J., "WIFI and Magnetic Fingerprint Positioning Algorithm Based on KDA-KNN", Proceedings of IEEE Chinese Control and Decision Conference, pp. 5409-5415, China, May, 2016.

[23] Niu, J., Wang, B., Cheng, L., and Rodrigues, J.J., "WicLoc: An Indoor Localization System Based on WiFi Fingerprints and Crowdsourcing", Proceedings of IEEE International Conference on Communications, pp. 3008-3013, UK, June, 2015.

[24] Ding, G., Tan, Z., Zhang, J., and Zhang, L., "Fingerprinting Localization Based on Affinity Propagation Clustering and Artificial Neural Networks", Proceeding of IEE Conference on Wireless Communications and Networking, pp. 2317-2322, China, April, 2013.

[25] León, O., Hernández Serrano, J., and Soriano, M., "Securing Cognitive Radio Networks", International Journal of Communication Systems, Volume 23, No. 5, pp.633-652, Wiley, 2010.

[26] Zhang, W., Liu, K., Zhang, W., Zhang, Y., and Gu, J., "Deep Neural Networks for Wireless Localization in Indoor and Outdoor Environments", Neurocomputing, Volume 194, pp. 279-287, Elsevier, 2016.

[27] Breiman, L., "Random Forests", Machine Learning, Volume 45, No. 1, pp. 5-32, Springer, 2001.

[28] Yang, S., Dessai, P., Verma, M., and Gerla, M., "FreeLoc: Calibration-Free Crowdsourced Indoor Localization", Proccedings of IEEE International *Conference* on Computer Communications, pp. 2481-2489, Italy, April, 2013.

[29] Wu, C., Yang, Z., Liu, Y., and Xi, W., "WILL: Wireless Indoor Localization Without Site Survey", IEEE Transactions on Parallel and Distributed Systems, Volume 24, No. 4, pp. 839-848, 2013.

[30] Yasmine, R., and Pei, L., "Indoor Fingerprinting Algorithm for Room Level Accuracy with Dynamic Database", IEEE 4th International Conference on Ubiquitous Positioning, Indoor Navigation and Location Based Services, pp. 113-121, 2016.