DOI 10.1186/s41070-017-0015-z

Scientific Phone Apps
and Mobile Devices

SOFTWARE ARTICLE Open Access

RSSI based adaptive indoor location tracker

Deepti Maduskar* and Shashikala Tapaswi

Abstract

Due to the impracticality of the Global Navigation Satellite Systems (GNSS) in the indoor locations, Wireless Network based Systems (WNS) are widely used for the indoor location estimation techniques. Compared to various other localization methods, the Received Signal Strength Indicator (RSSI) based techniques are highly appreciated because of their minimal complexity. Thus based on these, a generic application has been developed to track the location of any person in any area, with adequate Access Points (AP's), by comparing the Received Signal Strength (RSS) at his/her position with the already stored set of RSS from the dataset. To find out the best possible match from a group of possible locations in the dataset, the Root Mean Square Error (RMSE) method has been used, which has less complexity and more accuracy over its alternative methods. The unique features of this application includes the dynamic database which can be conveniently updated whenever needed, live input of any indoor location map through camera and immediate creation of a new dataset, allowing the application to be used by any person in any area.

Keywords: Global navigation satellite system, Global positioning system, Wireless network based, Received signal strength indicator, Root mean square error, Access point, Wi-Fi

Background

All the current Location Based methods are either Global Navigation Satellite System (GNSS) or Wireless Network Based (WNS). The Global Positioning System (GPS) can offer an accuracy of a few meters. Although the accuracy can be improved, the major disadvantage is its impracticality in enclosed environments, because it requires an unobstructed line of sight between the device and combination satellites.

Access Points (AP's) in a Wi-Fi network can be used to calculate the position of a Mobile Device (MD). Systems using Wi-Fi signals have achieved great significance as they provide the flexibility of calculating the position of a mobile using an already-existing infrastructure, implying no installation of hardware is required on the Mobile devices.

By measuring the Received Signal Strength (RSS) from various AP's at a given location, the exact location of a person inside huge premises like university campus or shopping mall can be estimated.

Broadly, there are two basic ways of utilising the Signal Strengths (SS) for position estimation algorithms. The first way implies the conversion of the RSS measurements into distance measurements, which are then fed as input for the location estimation algorithms. This approach can be termed as distance-based approach. For instance, in (Kotanen et al. 2003), first 802.11 Received Signal Strength Indicator(RSSI) measurements are converted into distance estimates using a simple radio wave propagation model, then extended Kalman filtering method (Paul and Wan 2009) is applied for computing position estimates from these distances. Similarly, in (Robinson and Psaromiligkos 2005) the distances are computed using a combination of maximum likelihood equations and a probabilistic signal propagation model. These are then fed to the nonlinear multilateration equations (Wang et al. 2009).

The second way, location estimation techniques consider RSS measurements directly as input. This approach can be termed as a power-based approach. The wireless indoor positioning system RADAR (Bahl and Padmanabhan 2000) provides a very good example for the power-based approach. Specifically, the location which

*Correspondence: sonicarmel@gmail.com
Atal Bihari Vajpayee Indian Institute of Information Technology and Management, Gwalior, 474015 M.P, India

has the minimum Euclidean distance between the measured signal strengths from three base stations and a previously recorded set of RSS measurements. In (Masuoka et al. 2010), the node to be located measures the Received Signal Strength from beacons, the nodes whose position is already known. That position is chosen which best satisfies a specified criterion (e.g., largest signal strength). The proposed work is based on this technique.

For the estimation of the user's position, the Nearest Neighbors in Signal Space (NNSS) algorithm is used for computing the distance of signal space between the observed and recorded SS (Bahl and Padmanabhan 2000). The nearest neighboring point will be considered as the user's position. The advantage of this approach is that it can reduce multipath problems (Sayed et al. 2005; Liu et al. 2007). The Root Mean Square Error (RMSE) (sometimes also referred to as the Root Mean Square Deviation, RMSD) is used as the distance measure (Bianchi et al. 2007). The RMSE serves to aggregate the magnitudes of the errors in predictions for various times into a single measure of predictive power. The reason of using this method in the app is the slight difference of the RSSI values obtained over time for the same position. So using this method we can find out the nearest position based on the RSSI values that gives the least RMSE. Although Euclidean distance can also be used for the same purpose but Euclidean distance is generally used to show the distance between two points whereas the RMSE is used to show the deviation between the two points. The application needs to calculate the deviation between the observed and recorded RSSI values. Therefore the RMSE value is more suitable for such calculations. The proposed work is a practical approach developed for tracking a person's movement in a known location. The android app also helps a person to locate himself in an indoor scenario and thereby navigate to his/her destination. The app generates a record file that contains the coordinates of the locations travelled by a person in the layout. The location estimation is done by measuring the RSS of Wi-Fi signal from the various AP's. It is basically a generic android application which can be installed on any mobile device and server. The app even allows a user to take any location with adequate AP's as the tracking background, build his/her own dataset and start the tracking process.

The rest of the paper is organized as follows: "Related work" section provides a glimpse of the literature review on the current existing location estimation techniques. "Proposed approach" section gives a detailed discussion on the proposed algorithm and the designed application layout. "Result and discussions" section provides the experimental results and the error estimates in various scenarios and analysis based on the results. "Conclusion" section finally concludes the research work with the scope of future work.

Related work

Given the insufficiency of the GPS systems, the use of RSS for tracking purpose is quite prevalent. A lot of work already has been done in this field. Toloza et al. (2014) has done much in this area by contributing in tracking by determining the position using the fingerprint algorithm. In the Fingerprint algorithm, first a radio map is blueprinted containing RSSI measurements for each AP under the coverage area of the MD. After all values have been obtained, the mode or average is calculated with the purpose of grouping the values, obtaining a strengths vector for each sampling point. These vectors are stored in the Fingerprint database. Thus the location of any device can be determined by comparing the strengths vector obtained against the vectors stored using the Euclidean distance. In (Chuenurajit et al. 2012), the same work is done using a min-max estimation, while min-max estimation provides a high level of accuracy but the complexity of the algorithm is very high. The proposed architecture has used a more proper estimation technique of RMSE. There are various researches that have been carried out in this field such as (Wei et al. 2014) but they are mostly hardware based where the algorithms are compatible within the AP's but the proposed approach has overcome these issues by a more generic app which can be installed in any android mobile operating system and applied on any AP in any area.

Proposed approach

Experimental testbed

The experiment consists of two testbeds spread in an area of 1200 m^2, both located in the C-block of ABV-IIITM. Testbed-1 is on the ground floor of the block whereas Testbed-2 is on the first floor of the block. Both the testbeds were fitted with maximum of 4 AP's, which may be functional or dis functional, according to the need. The AP's are evenly distributed as shown in Fig. 1.

Data collection

The application requires a storage of Signal Strength (SS) from various locations onto its database. This data can be collected using the same app by physically going to various locations and choosing the "create new dataset" option in the application. By specifying a server address, the data is sent to that particular server which appends these values in a text file which can be named according to the user's preference.

Location tracking

After the dataset has been built, the user can move to any random position and press the "scan" button. It scans the Received signal strength at his/her location from various AP's. The scan gives a set of (SSID,RSS) pairs of values from the AP's which are in range of the device.

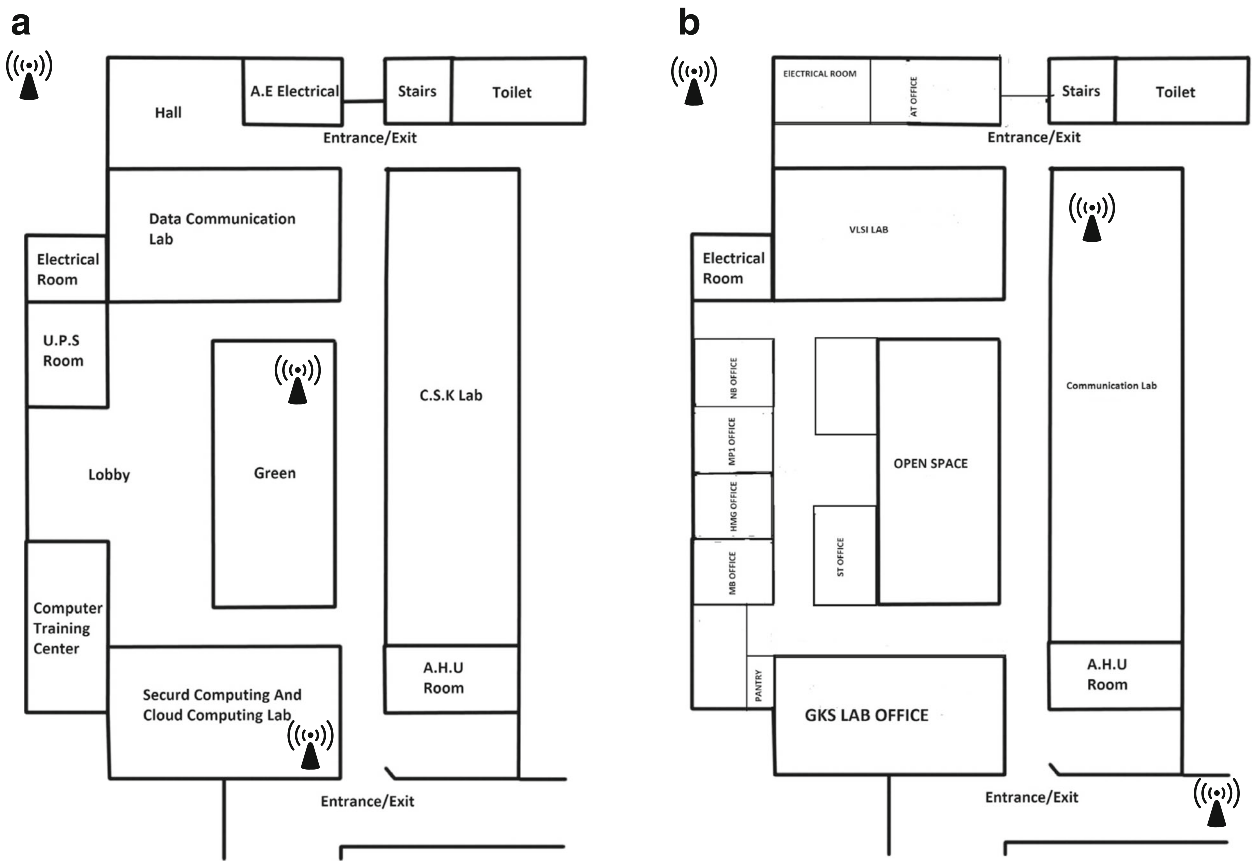


Fig. 1 a Distribution of AP's in testbed-1 **b** Distribution of AP's in testbed-2

The principle behind finding the coordinate inside a layout is to find out a set of RSS values which are most similar to the RSS values obtained during a scan for location estimation. This is done by finding the RMSE between the values obtained from scan and each tuple in the dataset. The value in the dataset which gives the least RMSE is said to be the one most similar to the value obtained in the scan. Thus we can use that corresponding coordinate value in the dataset, thereby estimating the position.

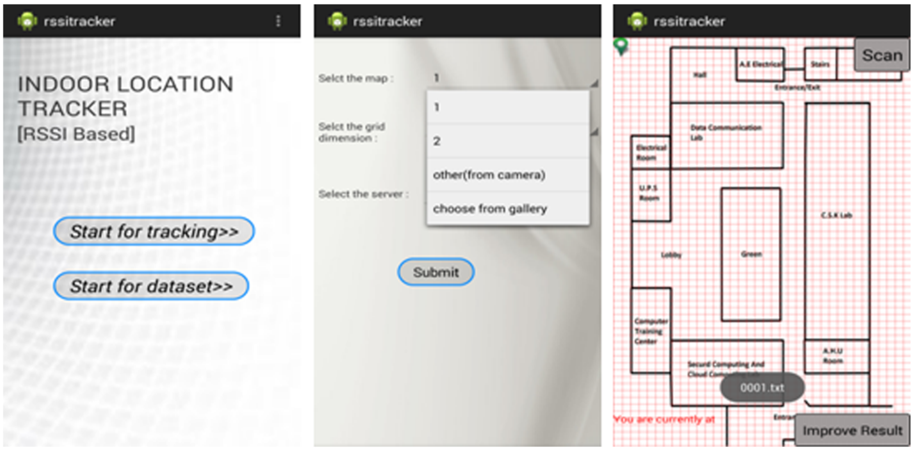


Fig. 2 Mobile user interface of the application

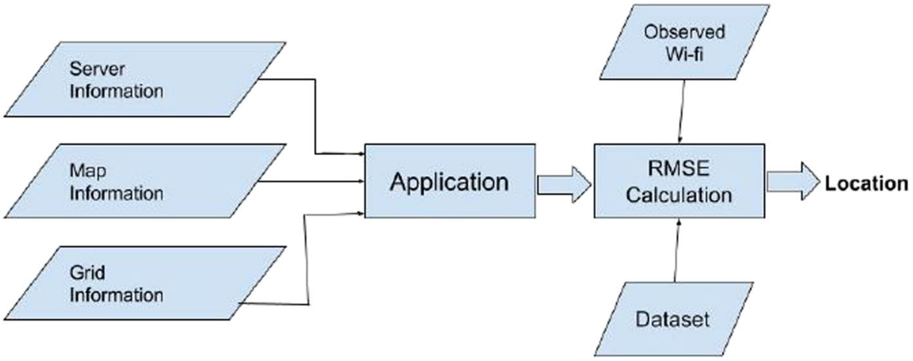


Fig. 3 Position estimation diagram

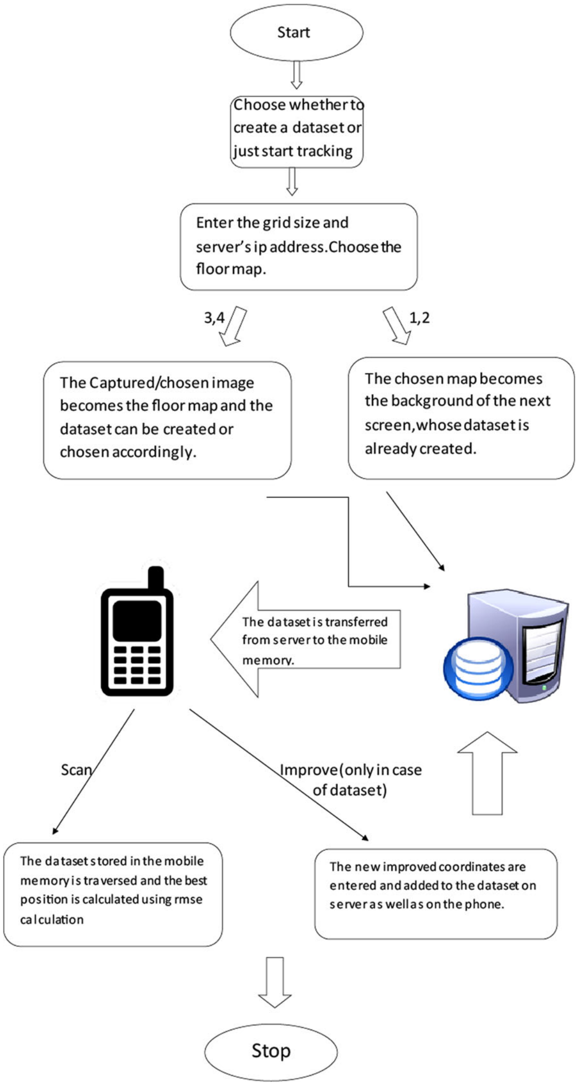

Start
Choose whether to create a dataset or just start tracking
Enter the grid size and server's ip address.Choose the floor map.
3,4
1,2
The Captured/chosen image becomes the floor map and the dataset can be created or chosen accordingly.
The chosen map becomes the background of the next screen,whose dataset is already created.
The dataset is transferred from server to the mobile memory.
Scan
Improve(only in case of dataset)
The dataset stored in the mobile memory is traversed and the best position is calculated using rmse calculation
The new improved coordinates are entered and added to the dataset on server as well as on the phone.
Stop

Fig. 4 Schematic diagram

The RMSE of a RSSI value with respect to the estimated RSSI value is defined as the square root of the mean squared error:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \left(S_{obs,i} - S_{model,i}\right)^2}{n}} \tag{1}$$

where $S_{obs,i}$ is the observed RSS and $S_{model,i}$ is the stored RSS at a location i. This is done with all the tuples and the tuple which gives the minimum value id chosen as the estimated location. If the estimated location is not satisfactory, the "improve result" button can be pressed and the new coordinates can be entered. Due to the dynamic nature of the dataset, this tuple will be appended. This increases the accuracy of location estimation.

Implementation

There are basically two options on the introduction screen of the app as shown in Fig. 2:

- Dataset.
- Tracking.

Basically by choosing "Dataset" option, it is required to choose a layout, build and/or improve its dataset, using different grids sizes. It is supposedly for the core administrators. Whereas choosing the "Tracking" option allows to load the defined dataset and start calculating the location, basically for the end users.

After choosing the mode of usage, the second screen has input fields for the layout, grid size (only in case of dataset mode of usage) and server's ip address. There are basically four options for the layout:

- 1: Loads the predefined layout of the testbed-1.
- 2: Loads the predefined layout of the testbed-2.
- 3: Allows to capture a live picture of the internal layout of any area and build a dataset (only in dataset mode of usage) or choose an already built dataset.

– 4: Allows to choose the layout from already stored pictures on the phone and build its dataset (only in case of dataset mode of usage) or choose from the existing datasets.

The chosen layout is set as the background of the next screen. The chosen dataset is loaded onto the phone's memory, from the server using a simple php script. In case of "choose dataset" option, a new dataset is built on the server and loaded onto the phone's memory, using a simple php code.

The next screen has a button "scan" which scans the current RSSI values of all the AP's and compares them to the nearest values in the dataset using RMSE method and calculates the nearest position. In case of dataset mode of usage, an additional button is provided i.e. "improve" button. Given the dynamic nature of the environment, the RSSI values change and thus the dataset should be refined with more and more possible data tuples. Therefore the new improved coordinates can be entered in the alert box, after clicking on the button. The new coordinates get written on the dataset on both the server and the phone's memory, by running a simple php script on the server and in the app.

Proposed architecture

The schematic diagram of the proposed algorithm is shown in Figs. 3 and 4.

Result and discussions

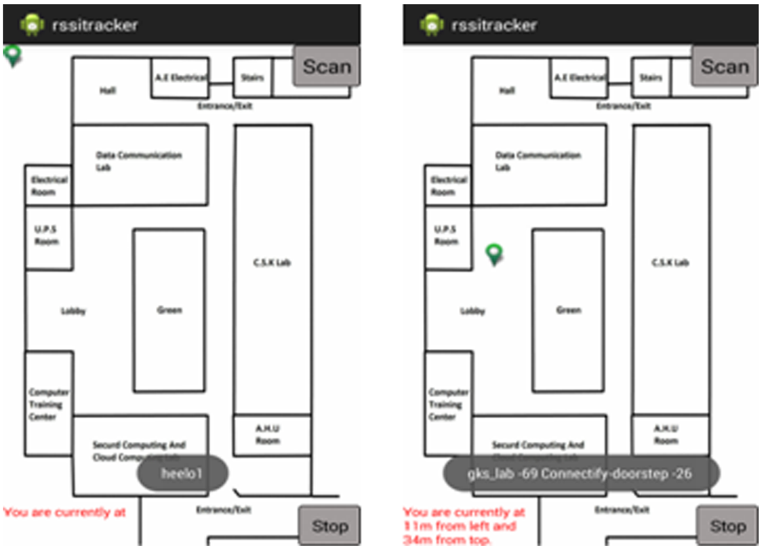
For experimentation, the dataset for testbed-1 and testbed-2 has already been built and stored on the server. The dataset for testbed-1 had a total of 10,000 tuples whereas the dataset for testbed-2 had a fewer tuples (1500) compared to testbed-1. Both the testbeds were tested with 4, 3 and 2 AP's. All the AP's were evenly distributed across the area for more authentic RSS values for each and every point. Each point represents a 1 × 1 grid corresponding to a 1m × 1m area of the testbed.

Figure 5 provides the graphical result as shown by the app after the location has been tracked.

Table 1 shows the RMSE between the estimated position and the actual position of the tracking with different number of AP's. The mobile device was moved in the whole area and the X and Y coordinate of the calculated position were compared with the X and Y coordinate of the actual position. Since the area of testbed was big, it is difficult to show each and every location coordinate tracking in the table. A combination of entries for depicting results effectively were taken to show the pattern and make the conclusions. The dominating entries having maximum error as well as minimum error have been shown such that the conclusions could be drawn in an unbiased manner.

Thus according to Table 1 we can infer the following:

1. The size of the dataset has a huge impact on the accuracy of tracking. Large number of tuples in the dataset results in more accurate location estimation. Thus the efficiency increases with increase in the number of tuples as shown in Fig. 6. As experimented, the time taken by the to compute the location does not increase significantly.
2. The number of AP's plays a major role in the tracking process. The relation between the number of AP's and the accuracy is significant. The accuracy increases with an increase in the number of AP's but



heelo1

You are currently at

gks_lab -69 Connectify-doorstep -26

You are currently at
11m from left and
34m from top.

Fig. 5 Tracking for the point [11,34] for testbed-1 with 2 AP's

Table 1 Tracking results for ground floor

Floor	No. of AP's used	Actual position [X,Y]	Tracked position [X',Y']	Error
Ground	4	[14,38]	[14,38]	0
	4	[7,6]	[5,6]	1.41
	4	[10,3]	[10,8]	3.53
	4	[14,7]	[12,10]	2.73
	4	[26,6]	[25,6]	0.70
	4	[11,34]	[11,34]	0
	4	[12,36]	[12,36]	0
	3	[14,38]	[14,38]	0
	3	[7,6]	[3,6]	2.82
	3	[10,3]	[10,1]	1.41
	3	[26,6]	[23,4]	2.73
	3	[12,36]	[12,34]	1.41
First	2	[14,38]	[14,38]	0
	2	[7,6]	[2,6]	3.53
	2	[10,3]	[10,1]	1.41
	2	[11,34]	[11,34]	0
	2	[12,36]	[11,26]	0.70
	4	[14,38]	[14,38]	0
	4	[7,6]	[6,6]	0.70
	4	[10,3]	[10,6]	2.12
	4	[14,38]	[12,38]	1.41
	3	[7,6]	[3,6]	2.82
	3	[10,3]	[7,3]	2.12
	3	[14,38]	[12,38]	1.41
	2	[7,6]	[5,2]	3.16
	2	[14,38]	[12,36]	2

after a certain point, the accuracy stabilises itself even on increasing the AP's.This is well shown in Fig. 7. Less number of AP's in an area will correspond to less number of RSSI values thereby low accuracy. This is shown in Fig. 7 (for 2 AP's). Whereas more number of AP's may result in phenomenon like masking. The mask effect occurs when one AP cannot be detected because its signal is covered by one or more of the other AP's. Thus there is a definite number of AP's required in a given area to accurately determine the location. Figure 8 shows the box plot of the standard deviation of the number of points accurately tracked for different number of AP's. The plot shows the mean of the combination of entries obtained after conducting multiple experiments.

3. Not only the number of AP's play an important role, but also the position of these AP's play a major role. The AP's must be distributed uniformly in the area.
4. The RSSI values are subjected to change due to various environmental factors leading to a decrease in the correctness of the model. Therefore the proposed approach provides a dynamic dataset which can be updated in constant intervals.

It is not enough to measure the performance of a positioning system only by its accuracy (Liu et al. 2007). The various other performance benchmarks used for observing the location system are complexity, robustness, scalability and cost. Since the application is a real time application, the computation time is minimal, despite of the large size of the database, due to the simplicity of the RMSE algorithm. High robustness requires the system to function normally even when some signals are not available or some signals are additional. The additional signals from the AP's which do not match with the AP's in the dataset are ignored from the calculations. Since the dataset is being built beforehand with the adequate amount of AP entries and can be updated in hand, missing AP's is not a problem. Thus the proposed algorithm accounts for robustness as well as scalability. Cost is the main merit of the proposed work. The application can be easily installed on any android operating system and no additional changes are required to be made on the existing infrastructure for the system to work.

Note:

- The SSID of the AP should not contain any white space.
- The name of the dataset should be entered such that it already exists,otherwise a new dataset will be built at both the server as well the phone.
- The resolution of the phone should be 480×854 pixels.
- The app should be run on the mobile connected to a proxy free Internet.
- The server should have a static ip (otherwise the mobile and the server must be in the same local area network).
- The app requires a working Internet throughout the time it runs.

Conclusion

Since the outdoor tracking systems have become quite successful and have achieved quite a good accuracy, the indoor tracking systems are still battling to achieve high accuracy. One of the ways of indoor tracking is by using the RSS obtained from the AP's. The proposed approach

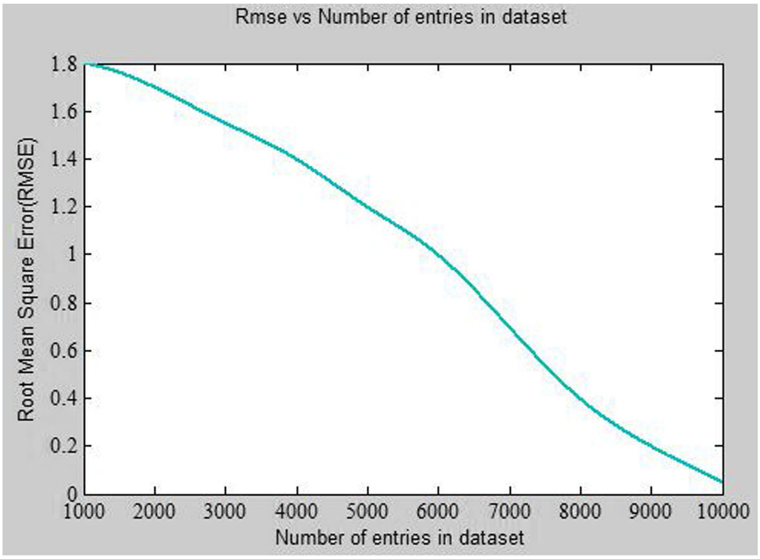


Fig. 6 Graph for RMSE v/s Number of tuples in dataset

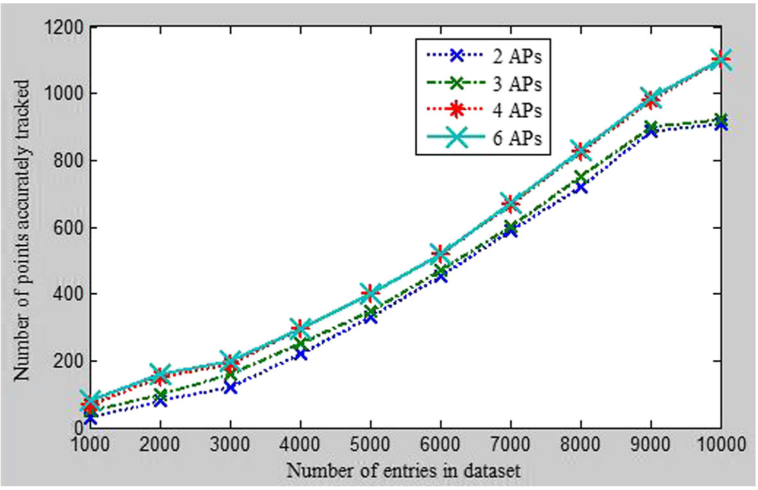


Fig. 7 Points accurately tracked v/s number of tuples in dataset for 2, 3, 4, and 6 AP's respectively

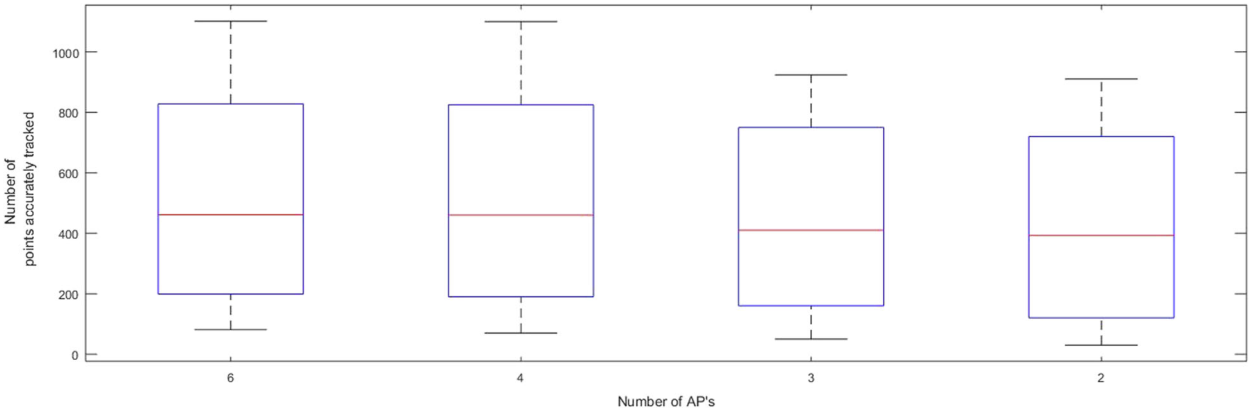


Fig. 8 Box plot showing the deviation in the number of points accurately tracked for different AP's

is an initiative in this direction to track an indoor location based on the RSSI and RMSE concepts. A generic application has been developed, which can be installed on any android platform and operated by server located at any part of the world. It allows the user to upload a layout from anywhere and build his/her own dataset.

The accuracy of the application is being tested by creating and comparing the dataset of two different areas with different conditions. The accuracy depends upon various factors like the number of tuples in the dataset, number of AP's, location of the AP's etc. Keeping in view the above points the accuracy of the position estimated is high in the proposed method compared to other methods. The app has been tested multiple times in a real time environment. Noticeably there was no significant impact of the moving objects on the observed tracked positions since the positions of the AP's remain fixed. The tracked position is slightly affected by the environmental factors and the error is in the permissible range. Interestingly one of the novelties of the application is its suitability for crowded enclosed environments.

The future work includes the increase of the accuracy of estimation by using a more precised error calculation tool.

Availability and requirements

- **Project name**: Indoor Location Tracker
- **Operating system(s)**: for server platform independent; for mobile devices : Android
- **Programming language**: Java
- **Other requirements**: Android development toolkit 22.3.0 - The toolkit contains eclipse 1.4.1, sdk -Software development kit, api level 19.

Authors' contributions

All authors of this research paper have directly participated in the planning, execution, or analysis of this study. Both authors of this paper have read and approved the final version submitted.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 24 December 2016 Accepted: 26 May 2017
Published online: 19 June 2017

References

Bahl P, Padmanabhan VN. RADAR: An in-building RF-based user location and tracking system. In: INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE Vol. 2. Tel Aviv: IEEE; 2000. p. 775–84.

Bianchi G, Melazzi NB, Piccolo FL. Impact of chosen error criteria in RSS-based localization: Power vs distance vs relative distance error minimization. In: Computers and Communications, 2007. ISCC 2007. 12th IEEE Symposium on MW-9. Las Vegas: IEEE; 2007.

Chuenurajit T, Suroso D, Cherntanomwong P. Implementation of RSSI-Based 3D indoor localization using wireless sensor networks based on ZigBee standard. J Inf Sci Technol. 2012;3(2):1–6.

Kotanen A, Hännikäinen M, Leppäkoski H, Hämäläinen TD. Positioning with IEEE 802.11b Wireless LAN In: Gong K, Niu Z, editors. Personal, Indoor and Mobile Radio Communications, 2003. PIMRC 2003. 14th IEEE Proceedings on. Beijing: IEEE; 2003. p. 2218–22.

Liu H, Darabi H, Banerjee P, Liu J. Survey of wireless indoor positioning techniques and systems. IEEE Trans Syst Man Cybernet Part C (Appl Rev). 2007;37(6):1067–80.

Masuoka R, Avancha S, Thakkar S, Agre J. U.S. Patent No. 7,751,829. Washington, DC: U.S. Patent and Trademark Office; 2010.

Paul AS, Wan EA. RSSI-based indoor localization and tracking using sigma-point Kalman smoothers. IEEE J Selected Topics Signal Process. 2009;3(5):860–73.

Robinson M, Psaromiligkos I. Received signal strength based location estimation of a wireless LAN client. In: Wireless Communications and Networking Conference, 2005 IEEE Vol. 4. New Orleans: IEEE; 2005. p. 2350–4.

Sayed AH, Tarighat A, Khajehnouri N. Network-based wireless location: challenges faced in developing techniques for accurate wireless location information. IEEE Signal Process Mag. 2005;22(4):24–40.

Toloza J, Acosta N, Kornuta C. WiFiPos: An in/out-door positioning tool. 2014. arXiv preprint arXiv, 1399–1402.

Wang X, Bischoff O, Laur R, Paul S. Localization in wireless ad-hoc sensor networks using multilateration with RSSI for logistic applications. Procedia Chem. 2009;1(1):461–4.

Wei J, Ji Y, Yu C. Improvement of Software Defined Radio based RSSI localization with bias reduction. IFAC Proc. 2014;47(3):7164–9.