(Stable) Virtual Landmarks: Spatial Dropbox to enhance Retail Experience

Swadhin Pradhan, Ananth Balashankar, Niloy Ganguly, and Bivas Mitra
Department of Computer Science and Engineering
IIT Kharagpur, India

***Abstract*—Landmarks are signatures of our surroundings which help us to uniquely identify a location. Recent studies show that like us humans, it may be possible by the sensors on mobile devices to identify landmarks [1], [2]. This can open up the possibility of a lot of applications in the domain of augmented reality, gaming, retail etc. However, to make such applications a reality, a particular landmark need to be stable across mobile phones, persons carrying the mobile phones etc. This paper specifically builds up a framework to discover such stable landmarks and demonstrates its utility in the development of next generation apps. In order to identify such virtual landmarks, we employ a clustering algorithm to perform non-intuitive feature combination of sensors like Accelerometer, Gyroscope, Magnetometer, Light, Sound, Wi-Fi, GSM signal strength etc. Further, we rigorously test the clusters to ensure that landmarks are stable across different devices, people, and time. According to our results, change in device affects the stability of a landmark most. Finally as a proof of concept, we develop a prototype system *RetailGuide* using landmarks to facilitate smart retail analytics cum recommendation service.**

I. Introduction

In retail sector, delivering a smarter personalized shopping experience to the smartphone holding customers is one of the innovative strategies recently adapted and widely discussed. Innovative mobile apps like (say) *RetailGuide* can help in personalizing and enhancing individual shopping experience. Let us consider the following scenario to highlight the utility of the smart city apps like *RetailGuide*. Alice enters a neighborhood retail store to buy a dress. While moving towards the garment section, she checks into her app *RetailGuide*. This app enables her to browse few reviews by her friends about the jeans section of that very store. On her way to cash counter, while crossing the book section, she just receives a notification by *RetailGuide*. It is about a previously set reminder to buy the latest novel by Dan Brown. While leaving the store, she posts some comments about the not-so-good customer service of that store. Similar to the client like Alice, *RetailGuide* proves useful for the retail owner Bob. Bob is happy about giving reward points to a loyal customer. Moreover, he takes note about incompetent customer service of his store to make the store more customer-friendly. So, *RetailGuide* will be like Google Analytics in physical store space for Bob like owner.

Efficient deployment of *RetailGuide* app requires accurate micro-level identification of locations. One elegant way to perform micro level localization is to introduce *virtual landmarks* [1], [2]. The concept behind virtual landmark is the following. Thanks to the availability of the embedded sensors (accelerometer, gravity, gyroscope, magnetometer), the smartphones have the ability to recognize the ambience and behavior of users. Consequently, the smartphones can *listen* to the distinguishable environmental signatures to identify a given location. The places might be a corner of a corridor, a GSM blind spot or a specific Wi-Fi zone. We utilize these unique fingerprints of natural locations in smartphone sensor space, as virtual landmarks. These virtual landmarks can then be effectively used as dropboxes of comments by the *RetailGuide*. By dropbox we mean that these landmarks will be like spatially scattered information containers. User comments will be tagged with the nearest landmark i.e. a dropbox, to facilitate efficient indexing and retrieval of these comments in future. Moreover, shop owner also can put location specific offers to the intended dropbox, which will provide most relevant offers to any user.

In this paper, we propose a thorough virtual landmark enumeration procedure via clustering sensor data and evaluate our algorithm in an indoor space. It is important to note that, in order to effectively use them as dropbox, these landmarks need to be stable - that is, the the mobile phone based landmark signatures need to be trustworthy (a) across the different smartphones, manufactured by different vendors (b) across the different users. We carried a detail test of stability of landmarks across different factors and discovered interesting insights like device hardware specific heterogeneity mostly affects the stability. In order to demonstrate the proof of concept, we have developed a prototype*RetailGuide* app to identify the virtual landmarks. This app identifies and then utilizes these virtual landmarks as dropboxes distributed in physical space, where users can drop their comments about something nearby.

Figure 1 shows a concept image of a shopping area annotated with virtual landmarks which will help *RetailGuide* app. The figure shows different landmarks at different places of the shopping area, e.g. a magnetometer landmark nearby mobile section or a sound landmark near customer care. A few of the landmarks may be overlapped like Wi-Fi landmark (denoting a specific Wi-Fi zone) and Gyroscope - Accelerometer landmark near one book section. The shapes of the landmarks are shown as circular, but in reality it can take any geometrical shape under some area bound.

Summing up, the primary contributions of this paper are the following.
(a). Proposing a thorough virtual landmark pruning algorithm: We describe a methodology to detect virtual landmarks using adaptive clustering algorithm which is similar to the scheme proposed in [1] and validate its accuracy in an indoor

978-1-4799-3635-9/14/$31.00 ©2014 IEEE

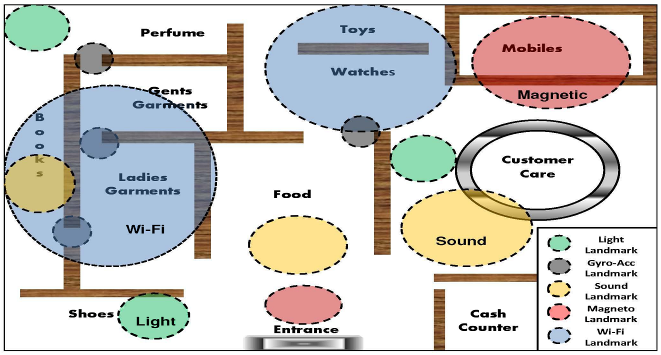
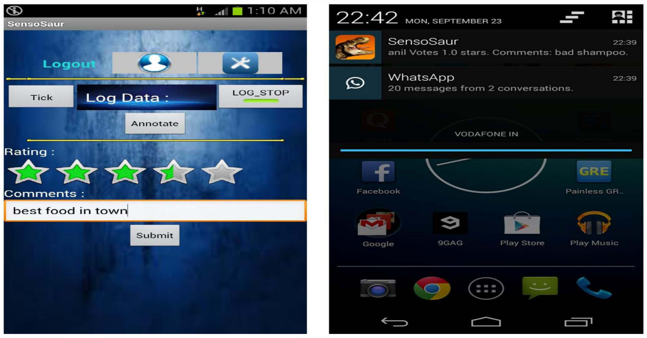


Fig. 1. Concept image of a landmark augmented shopping mall needed for *RetailGuide* Application.



Comments : best food in town

SensoSaur 22:39
anil Votes 1.0 stars. Comments: bad shampoo.

WhatsApp 22:39
20 messages from 2 conversations.

Users' Commenting Interface **Pushed Notification of Comments**

Fig. 2. Screenshots of *RetailGuide* Android Application. The pie chart shows the places where users have visited.

scenario.

(b). Studying the stability of virtual landmarks: We perform extensive experimental study with Samsung smartphones to know the internal dynamics of the stability of these landmarks and its dependence on different parameters like devices, time, and persons. We find their respective contribution on the instability of the landmarks and gain interesting insights.

(c). Developing an end-to-end prototype of *RetailGuide*: We build an end-to-end simple prototype of *RetailGuide* to test the feasibility of a system based on virtual landmarks and we test the workings of this prototype in an indoor space which mimics a shopping mall floor.

Section II focuses on the overview of the workings of a *RetailGuide* app based on virtual landmarks. Next, in Section III, we focus on the architecture of our landmark enumeration system, *Landmarker*. In the subsequent sections, we discuss the metrics, experimental setup, results, future works and related studies.

II. Overview of *RetailGuide* App

Our *RetailGuide* app caters to both user's and shop owner's need. User can comment via *RetailGuide*'s commenting interface and get relevant offers or recommendation via pushed notifications, as shown in figure 2. Shop owner can look into the detail analytics of his shop through the analytics window shown in figure 3. He can observe the upcoming trends, the customers' shopping patterns, combo offer suggestions etc, by just selecting appropriate section in the interface. For example, in figure 3, the pie chart reveals that buyers are more interested in Food and Utensils based on their movement patterns.

At the back-end, landmark enumeration service *Landmarker* collects the sensor readings and sends to the cloud. Sensor readings are location-stamped using a sophisticated dead-reckoning method. The processes in the cloud find the unique stable fingerprints via clustering, i.e., landmarks, mining the raw sensor data, as shown in figure 4. We create a stable landmark database in the cloud, as depicted specifically in Block 1 of figure 4. Interestingly, these landmarks also can help later to calibrate user's current position, as shown in Block 2 of figure 4. Moreover, a comment indexing and retrieval service also runs at the back-end to tag users' comments with nearest landmark and accordingly store them in the cloud. If a user searches for any review pertaining to any section of the store, this service helps app to return the relevant comments only by searching the nearest landmark, as shown in Block 3 of figure 4.

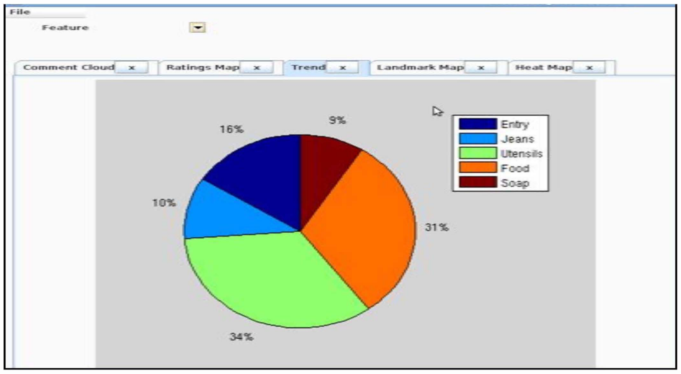


Fig. 3. Screenshot of Analytics Interface of *RetailGuide* Java Application.

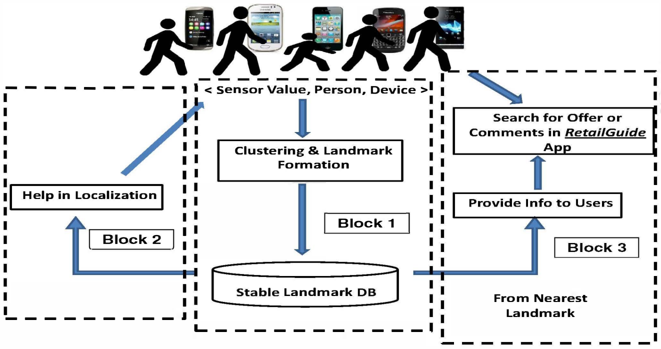

< Sensor Value, Person, Device >
Clustering & Landmark Formation
Block 1
Stable Landmark DB
Help in Localization
Block 2
Search for Offer or Comments in RetailGuide App
Provide Info to Users
Block 3
From Nearest Landmark

Fig. 4. Concept image of architecture of *RetailGuide* Application based on landmarks. Block 1 represents the basic landmark enumeration component; Block 2 shows the feedback loop of localization correction using landmarks; and Block 3 depicts the commenting and review finding component

Obviously, the success of this kind of system totally depends on the availability and stability of landmarks. Our intuition says that different places, especially indoors, would have sufficient number of landmarks.

III. Architecture and Methodology

We start out with the high level organization and working of our system of landmark pruning service, i.e., *Landmarker* and later delve into the design details of each of its components. This is the core component of *RetailGuide* app.

A. Brief Description of Design Details of Landmarker

Landmarker service, running in the background of *RetailGuide* app, collects sensor data from different users and sends data to the cloud server for further processing. In cloud, we get the sensory landmarks via clustering the processed sensor data. We consider only those landmarks which recurrently occur for different traces. Finally, we store the stable landmarks systemically in a cloud database.

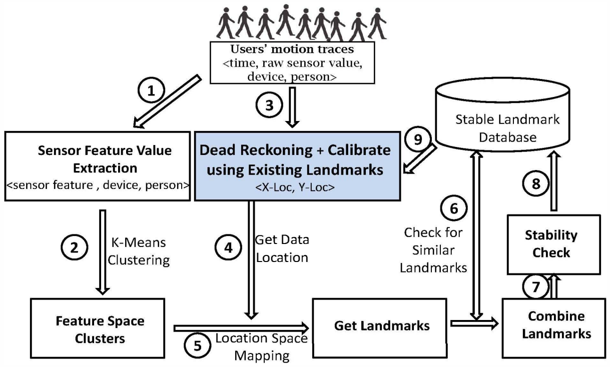


Fig. 5. Architecture of Landmarker : the landmark pruning system

Architecture Overview The overall architecture of our landmark enumeration system, *Landmarker* is shown in figure 5. Initially, the sensor data are collected from different devices and specific features are extracted after proper sampling and noise removal, as given in step 1 of figure 5. Thereafter, we cluster the sensor data in higher dimensional feature space using k-means algorithm. Next, we map the clusters in location space, using the dead reckoned location estimate of the data members, as shown in steps 3 and 4 of figure 5. Afterwards, we get the landmarks from these location clusters. Finally, we get the stable landmarks after combing through different traces and storing them in a database (steps 6,7 and 8 of figure 5). Interestingly, after bootstrapping stage, we can use these initial landmarks as minimizing the error of location estimates, which in turn helps to find more stable landmarks later. This iterative loop is initiated through step 9 of figure 5.

The different process components of the architecture shown are explained in detail in the following paragraphs.

B. Sensor Data Collection

To form the landmarks, we need to get the raw sensor data from different sensors of various devices carried by different users. Each collected sensor data tuple can be represented as $<$ Time stamp, Sensor Value, Device ID, Person ID $>$. The sensor data collection of *Landmarker* system can be divided into the following subprocesses:

(a) Data Sampling: We have collected the data using an android app through different android mobile devices. The sampling rate can be fixed to the highest value. But, android OS does not poll the sensor readings at regular interval, rather only records the data if any change in sensor value occurs. So, the frequency of unprocessed sensor data vary widely, e.g. very high for accelerometer and very low for GSM chip. Therefore, the sensor data is sampled at a constant rate of 50 Hz for uniform analysis. Then its noise is removed by passing it through a low-pass butter-worth filter. The data is then normalized according to a range of [-1,1], to have an uniform scale for clustering in feature combination scenario.

(b) Dead Reckoning: The raw sensor data tuple does not contain any location space co-ordinate. But, forming landmarks from clusters, these co-ordinates are necessary. For this purpose, we use the method of dead reckoning. It helps to trace an approximate path taken by the user from the accelerometer, gyroscope and compass readings. If it were a robot or car, we could have integrated acceleration twice and got the distance. But this leads to huge error in the case of smart phones, as shown in [1].

The mobility trace is best modeled by using a pedometer algorithm which counts the steps taken by the user. This is done by finding the peaks in the accelerometer-z data and calculating the stride length based on the number of steps taken per unit time [3]. We have used the method of dynamic time wrapping as discussed in [4] for removing noise and false peaks. We get the direction of the motion by reading the compass readings provided by the phone. However, this can be affected by the magnetic fluctuations in the indoor environment. This noise can be removed by opportunistically comparing with the angle calculated from the gyroscope readings and removing the extra bias as described in detail in [1]. Thus we add relative (x,y) co-ordinate to the sensor data tuple.

C. Sensor Feature Extraction

We then extract the features from the sensor data tuples that are used to cluster in order to obtain the landmarks. The features we have selected are given in table I. These are based upon previous works in the field of activity recognition. For each of this feature, we have taken the standard statistical measures - standard deviation and mean.

Below we give a brief description of a few features.

TABLE I. Features selected for Landmark Identification

Sensor	**Feature**
Accelerometer	$\sqrt{acc_x^2 + acc_y^2 + acc_z^2}$, $\lvert linearAcc_x\rvert + \lvert linearAcc_y\rvert + \lvert linearAcc_z\rvert$ (SMA)
Magnetometer	$\sqrt{mag_x^2 + mag_y^2 + mag_z^2}$, $\frac{d}{dt}\,(mag_y)$
Gyroscope	$gyro_z, RotationMatrix_z$
Sound	in dB
Light	Intensity
Wi-Fi	Access Point Similarity Signature
GSM	Signal Strength

AP Similarity Signature : The value of the access point similarity signature of Wi-Fi at two locations l_1 and l_2 is calculated as given in [1] by the formula

$$S = \frac{1}{|A|} \sum_{\forall a \epsilon A} \frac{min(f_1(a), f_2(a))}{max(f_1(a), f_2(a))} \quad (1)$$

where f_1, f_2 are the RSSIs of the APs at two locations l_1 and l_2 respectively and $A = A_1 \cup A_2$ denotes the total number of access points at two locations l_1 and l_2. This way, locations which have similar set of APs with approximately the same RSSI have low distance in the Wi-Fi feature space.

Signal Magnitude Area (SMA): SMA is defined as the acceleration magnitude summed over three axes within each window normalized by the window length [5]. It is an efficient depiction of the energy of motion.

In addition to these ten features with the corresponding statistical ones, we have also analyzed the combination of these features.

D. Feature Based Clustering

We now mine the features extracted from sensor data to find any unique characteristics. To do it in an unsupervised manner, we employ feature based clustering method on this processed sensor data. We have chosen k-means clustering for this, as this has been proven to be robust and widely used in practice. In this clustering, different dimensions are the features of different sensors. However, we have taken the following two options for implementing k-means :

Selection of k : We chose k by following the methods given in [6] where the optimal k are chosen by clustering random samples of the data for different k and choosing the one for which the intra-cluster centroid distance is minimized.

Selection of initial seed : The initial seeds are chosen by clustering over random samples of data and choosing the centroids of the cluster which performs best according to the distance metric given in [7].

But, not all clusters identified from k-means are candidates for landmarks. The clusters which are dense and which can distinguish from its neighborhood clusters are chosen. This is measured by the low average intra-cluster centroid distance and by the high average inter-cluster centroid distance. So, a dense small-area cluster would be good candidate of a landmark if it differs well from other clusters spatially. The properly normalized feature clusters, by taking note of the dimensionality, which satisfy these threshold conditions, are then passed on for location space mapping. We have also tried with different other clustering algorithms, which either yielded similar results like EM algorithm, or did not suit the purpose like hierarchical clustering, DBScan etc.

E. Clusters to Landmarks

Once we have these clusters, we check if they transform into spatial landmarks. For this we map each of these cluster-points in the feature space to location i.e (x, y) coordinates, as show in figure 6. These location points are now clustered using our augmented k-means algorithm. The clusters thus formed contain points which are near in both the feature space and location, thus an ideal candidate for a landmark. In figure 6, some of the clusters in feature space got scattered in location space. Thus, they are not considered as landmarks. We have considered different area thresholds for the landmarks formed in location space.

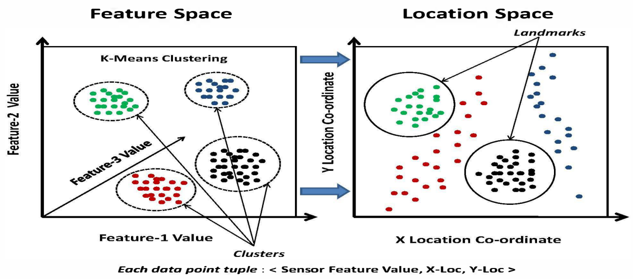


Fig. 6. Landmark Formation : Feature space to Location space Mapping

F. Combining Landmarks

Once the landmarks are identified for a mobility trace, their stability needs to be analyzed. This can be done by combining similar landmarks from different traces. The traces are logged by changing devices, person, and time. The convergence of the location of the landmark is dependent on the estimated landmark location of new traces. If similar landmarks from different traces are within a threshold distance, they are combined to give a new landmark, as shown in figure 7. We combine the corresponding points of the two landmark clusters and take an average of these two points. The basic assumption is that errors produced in different samples are independent. After this, we discard the points which do not fit in the landmark area threshold.
The landmarks thus identified are said to be stable if these occur recurrently in different heterogeneous traces. The stability of a landmark is now measured by the number of samples in which this landmark was encountered and is known as the confidence count of the landmark. We consider landmarks which appear in more than 50% of the samples as stable landmarks.

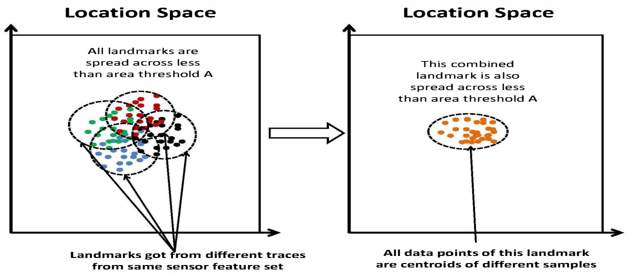


Fig. 7. Combination of landmarks

G. Calibration of location of sensor data

As we have discussed in the above section, simple dead reckoning based location estimation will lead to error. So, after a little bit of bootstrapping to find an initial set of landmarks with estimated location, we will bring them in the loop to estimate current sensor data location. We use the concept of Simultaneous Localization and Mapping (SLAM) [8] here to reduce the error in location. The correction is done by shifting the subsequent location points by the difference in the two (previous and current) estimates of the landmark location. Note that this assumption helps to make the system recursive and prevent advancement of location error. So, we can use them as location check points to fix the error. Moreover, this calibration step helps the system to converge to a stable set of landmarks with accurate location.

IV. Metrics of the System

In order to discuss the results and core issues of the problem, we would like to introduce a few simple metrics to detect and characterize virtual landmarks.

(a) **Area threshold for landmark -** This is defined as the area covered by a particular landmark. Default landmark area is chosen as 4 m^2, unless specified otherwise.

(b) **Feature Space Nearness -** It is the measure to determine whether two points belong to the same cluster. That is, given a cluster which has been built considering some sensor features, all pairs of points in that cluster need to be less than the specified threshold – which is termed as *Feature Space Nearness*. The higher the value of feature space nearness, the closer the data points are in feature space.

(c) **Confidence count -** It is the number of path traces in which a landmark is found, for example, if N traces are considered, confidence count N/2 means that the landmark has been detected in at least N/2 traces. The higher the value of the confidence count corresponding to a landmark, the higher its probability of being stable.

V. Experimental Setup

We conduct our experiments by collecting human motion traces with smart phones in user's hands. We have used Samsung Galaxy S2 I9100G and Samsung Galaxy S3 I9300 for our purposes. These phones provide us with sensors such as accelerometer, gravity, gyroscope, magnetometer, orientation, sound, light, Wi-Fi and GSM. Both of the phones are upgraded to android 4.1.2 (Jelly Bean). In our experiments, we collect these sensors' data while walking in the corridors with the phone held in the hand, facing upwards. We use *RetailGuide* app [9] with *Landmarker* service running in the background to conduct our experiments.

The data recorded internally is sent to the *RetailGuide* server. The server side code is written using php and MATLAB, and implements the dead reckoning, clustering, and landmark signature-matching algorithms. We assume constant orientation of the phone for easy understanding of the setup [4].

In order to understand the stability of these sensor landmarks and usability of *RetailGuide* app, we performed experiments where the user traces the same path multiple times. We covered 500 m^2 area in indoor environment (corridors of the Department), figure 8, as it best mimics the shopping mall scenario. To test the robustness of *RetailGuide* app and the backbone landmarks, data was collected multiple times on two devices, at two different times of the day - *morning* and *night* by four volunteers.

The default values of landmark area threshold, feature space nearness value and confidence count are taken as 4 m^2, 0.7, and N/2 (where N is the number of traces) respectively.

VI. Experimental Results

In this section, we first evaluate the potential of discovering landmark using mobile phones and then check the efficiency of *RetailGuide*.

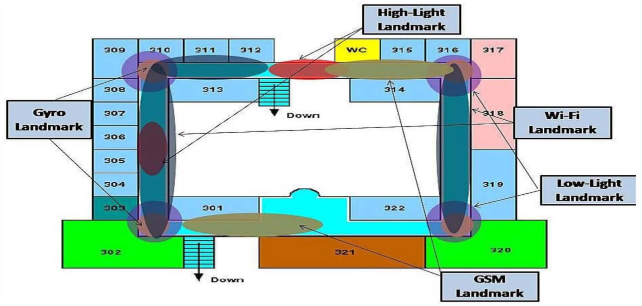


Fig. 8. Landmark annotated Indoor area map. This is the map of Second Floor, Computer Science and Engineering Building, Indian Institute of Technology, Kharagpur.

A. Feature Combination

In order to discover landmarks, as mentioned in Section II, sensor data from nine sensors are collected. For each sensor, several statistical measures like mean, standard deviation, kurtosis etc. are collected. Therefore each location is characterized by f (= *no. of sensors* $\times$ *statistical measures*) number of features. A group of points may get clustered based on a subset of features – therefore in order to find the best clustering condition one has to exhaustively look into all the subsets which would be 2^f - 1. This would explode the feature space and hence optimal clustering may not be feasible. But, interestingly we have found that we do not have to consider all the subsets. If we consider only two or three features in the combination, it will suffice our purpose. Both the count of clusters as well as landmarks (co-located clusters) decrease with the increase of combination counts, as shown in figure 9 and figure 10.

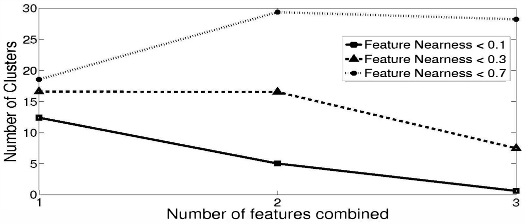


Fig. 9. Average number of clusters in our experiments. The increase in the number of clusters for parameter nearness 0.1, is due to too much relaxation of the constraint. We are taking too many insignificant clusters into account.

For experiment purpose we have taken only the *mean* of all the sensor data and combined different sensor's data to discover clusters. In figure 9 it is seen that the number of clusters formed decrease with the increase in the number of features combined for clustering. This is counter-intuitive because there are more number of possibilities for clustering, for example, for ten features, forty five ($\binom{10}{2}$) for two sensor data combination, one hunderd and twenty ($\binom{10}{3}$) for three combinations. Moreover, we also see similar trend in the case of stable landmarks, as shown in figure 10. Most importantly,

we see a dramatic decrease in the number of landmarks when we combine two or three features. This means that there is no need to explore the combinations which comprise of more number of features.
Thus, we have considered only one, two and three feature combinations of the ten features given in table I with their corresponding statistical measures (mean or variance).

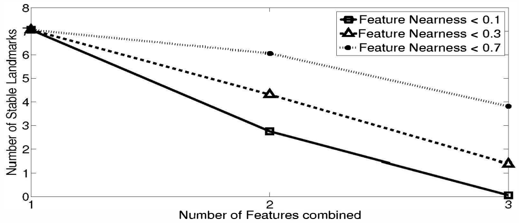


Fig. 10. Average number of landmarks in our experiments

B. *Effect of landmark area threshold on system design*

While testing with suitable area threshold for stable landmarks, we have observed an interesting phenomenon. Different set of sensor features are, in general, producing widely varying average landmark cluster area. For example, a Wi-Fi landmark cluster might cover an area of close to 30 m^2 whereas a light landmark will cover only a couple of square meters. Therefore, while choosing a sensor-spatial cluster as a candidate for a landmark, this threshold should ideally depend on which feature(s) it was clustered about. The cdf graph in figure 11 shows that various sensor features are clustered around different areas in the location space. The reason of this variation may be due each sensors different level of sensitivity to environment. Hence, instead of constant threshold, threshold need to be customized for each individual sensors.

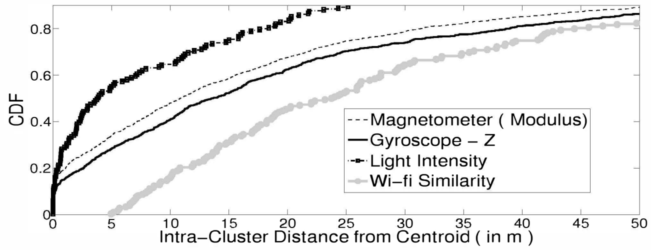


Fig. 11. CDF of landmark cluster area for different sensors

C. *Effect of Heterogeneity in the System*

We investigate the effect of heterogeneity on the stability of landmarks in this section. By stability of landmark in face of heterogeneity, we mean that landmarks are invariant in spite of changing the devices, the time frames or persons carrying the devices in different experiments.

(a) Person Heterogeneity: We felt that the variation of the walking style, movement speed of different persons can have an impact on the stability of landmarks. Therefore, we have conducted a small-scale experiment to collect traces with four persons. Figure 12 shows that the number of landmarks obtained by different users decreases as the confidence count of the landmarks increases. However, most of the users obtain roughly same number of landmarks. The number of landmarks obtained at confidence count N/2 is reasonable and on manual inspection are found to be of 'optimal' size (not too large or almost invisible). Hence N/2 is considered as default confidence count. The graph shown in inset of figure 12 shows that we are getting around 12 stable landmarks, which is considerably high for such a small indoor space. Each individual users besides discovering these stable landmarks also identify several 'unstable' landmarks.

(b) Time Heterogeneity : In this case, we have studied the effect of time of a day on the stability of landmarks. We have taken two time periods, i.e. day period (10 a.m. - 1 a.m.) and night period (8 p.m. - 11 p.m.), for collecting the traces using mobile devices. The intuition behind this experiment is that the signatures like light, sound etc. change with the time of the day, e.g., a busy shop becomes silent at night.
Although the count of the landmarks does not vary much; unlike previous case, the comparison for actual landmarks in figure 13(a) reveals that approximately 33% of the landmarks are stable. It is lower than the case of different users.

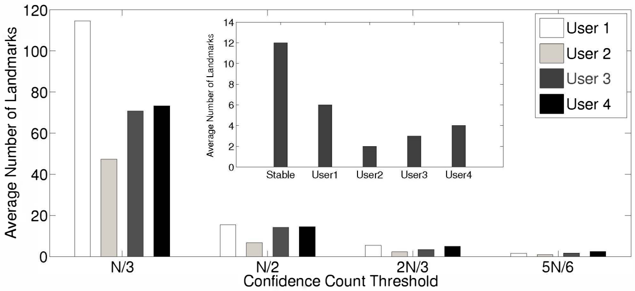
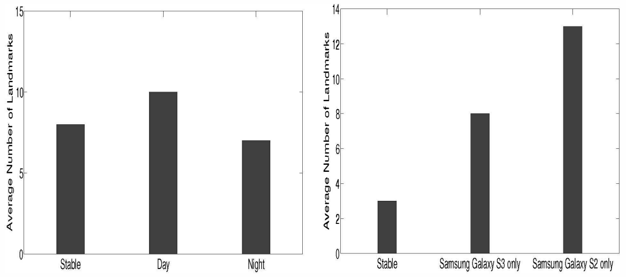


Fig. 12. Number of landmarks for different users. Here, N is the number of traces. That means each of the user has moved N number of times in the designated area. Inset figure shows the comparison of number of stable landmarks and user specific unstable landmarks

(c) Device Heterogeneity : We have repeated the experiments of collecting traces with two devices, namely Galaxy S3 and Galaxy S2, to test the effect of change of device on the stability of a landmark. Figure 13(b) shows that we get around 3 stable landmarks in this indoor space, which is considerably less than the earlier two cases. So, we can conclude that the effect of device heterogeneity has the most impact on the stability of landmarks.



(a) (b)

Fig. 13. (a). Comparison of number of stable landmarks and time specific unstable landmarks (b). Comparison of number of stable landmarks and device specific unstable landmarks

It is interesting to note that even though both of the devices are from the same manufacturer and same series, there have been a considerable difference of the hardware, subsequently, the landmarks. So, the inherent difference of sensitivity and precision of different sensors has a telling impact on the stability of landmarks. On the other hand, the effects of change of time and persons, are significantly lower than the case of devices. Therefore, if we want to create a corpus of stable landmarks to augment the location based services, we have to organize it with respect to different class of devices or a set of sensors, as hinted by [10].

D. Analytics from RetailGuide

The task of *RetailGuide* is to properly identify retail space by running the background *Landmarker* service. We perform an experiment to test its performance. In the experiment, users roam around with smart phones running *RetailGuide* app in the department corridor, which mimics a shopping mall situation in a controlled manner. Corridor corners are named as different sections of a shopping mall like food, clothing, utensils, and cosmetics. Users also comment while moving and get relevant offers cum recommendations. In this experiment, we have considered around 10 landmarks, which we use as dropboxes of comments.

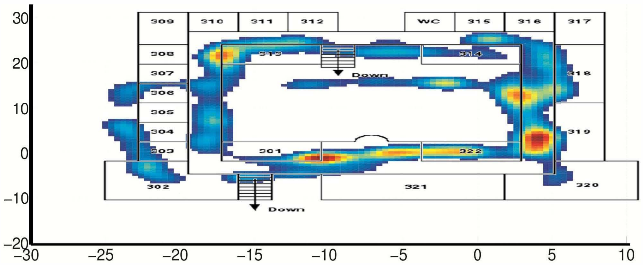


Fig. 14. Users' movement heat map found from *RetailGuide* Application.

A user's trail is inferred from his movement from the latest landmark. Figure 14 shows the heat map of users' movements in the corridor inferred from nearest landmark locations. Clearly, it contains some error as most of the users' movements are rectangular. From the estimation of the position of an user so derived, any comment she posts is tagged with that location by the cloud service. The service also accordingly attach this comment to the nearest landmark. The efficiency of the *Landmarker* algorithm would be measured in terms of the number of times it is attached to the correct landmark. Figure 15 illustrates example of comments posted by users at different locations. The circle shaped dots in the figure 15 denote correct location tagged comments and star shaped dots denote erroneous location tagged comments. In general, we get around 75% accuracy in attaching a comment to the correct landmark.

In both of the cases, we found considerable amount of error to predict the actual user trails or comment locations, due to variance of landmark formation from different smart phones. In order to understand the variance we need to study the nature of *localization error* occuring due to heterogeneity.

Localization Error due to Heterogeneity: We have created a stable landmark database using a specific triplet of < Device, Person, Time >. In order to understand the impact of an individual, the time and device, we change any one of these three parameters and test the deviation from ground truth (identified landmark), i.e. localization error. In figure 16,

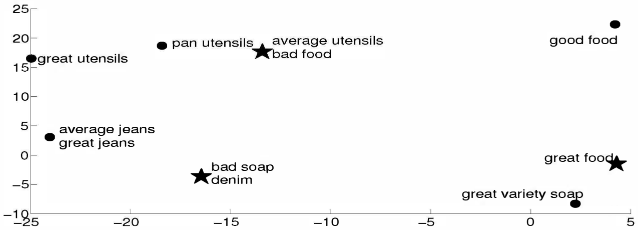


Fig. 15. Comments of different users through *RetailGuide* Application.

we can see that if we change device, person or time, the localization error will increase. However, the effect of device change on error is the most significant, which is in line with our previous findings.

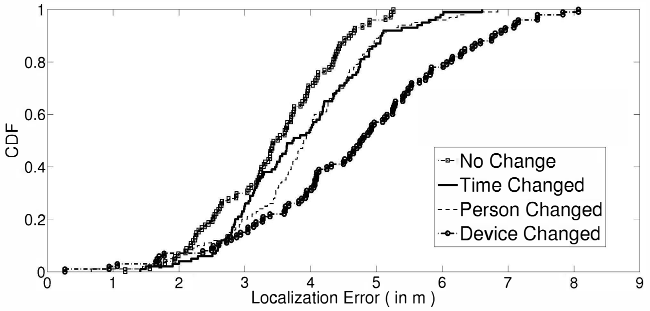


Fig. 16. Localization Error in Indoor Space

VII. Limitations and Future Work

We note down some of the limitations of the system which need to be tackled to make the system deployable.

(a) Experiment with more devices would help : One may ask we could have taken more devices to conduct an elaborate set of experiments. But, in this small scale experiment, we have considered the best case for devices by choosing same generation devices from same manufacturer. Even then, we have found that the device is the most prominent parameter affecting the stability of landmarks. A recent work [10] has shown that there are clusters in the sensors of different devices such as nokia, htc, iphone, lg handsets. As the effect of device is the most, we can have a set of landmarks belonging to each device class.

(b) Phone orientation : Since our study was limited to identify the impact of changing devices, time and user, we have neglected the impact of the phone's orientation as held by the user. This assumption is fine if we assume the user's mobility pattern is unaffected by the phone's orientation. However for example, a user might walk faster if his phone is in his pocket rather than held in his hand. But, we did not proceed further in this direction because a work by [4] has taken this into account to correct the dead-reckoning based on the phone's orientation as well.

(c) Ensuring privacy of users : This kind of pervasive applications generally suffer from privacy issues. In future, we would like to explore to address the zero sum game of privacy and usefulness of an application.

VIII. Related Work

The idea of landmark for navigation or localization is pretty ancient. From the pole star guiding the sailors to helping out today's busy teens to find the common meeting place, landmarks have always been integral to our daily life. Moreover, migratory birds find their winter abode [11], desert ants find their food [12], or honey bee tracks back their way back to home [13] using spatio-temporal landmarks. Even human minds keep track of some route or places in terms of landmark maps [14]. But, most of the implementations of current Localization based services (LBS) rely heavily on a GPS sensor to give the exact position of the mobile device the user is operating. However, the low accuracy (∼10m) and high power consumption of GPS are serious drawbacks given that we require high level of accuracy at low energy cost. On the other hand, for the indoor situation, GPS is almost completely unavailable.

Interestingly, this forced researchers to revisit the idea of landmarks for localization. We can find the essence of landmarks in recent ambience signature based localization works. Some of the recent localization or place recognition systems have been EZ localization [15], GSM signal fingerprinting [16], Surroundsense [17], RF based techniques [18] or Wi-Fi based schemes [19]. RF based or Wi-Fi based schemes either suffer from infrastructure dependence or high calibration time, while localizing places. On the other hand, a few works augment urban dead-reckoning [20] to improve indoor localization using mechanical sensors like accelerometer and gyroscope. However, these works mainly concentrate on the indoor localization, some recent works [21], [22] also use sensors in smartphone as an ally for outdoor localization also. These systems although depend upon the signature of surroundings, they do not explicitly bring the concept of landmark on board. Although distinguishing signature is the core of any landmark, landmark can be more than a vector of signatures. This idea of landmark for simultaneously localizing object is first explored by robotics community through the works of SLAM [8]. However, they are concerned about finding visible landmarks through costly sensors. Their goal was appeased easily as the mechanical movement of robots help them to do precise dead reckoning. However, the concept of invisible landmarks through the cheap smartphone sensors are brought forward by the authors of UnLoc [1]. They, like us, use different sensor signature to form landmarks to provide regular location fixes. But, they are only confined to the localization for their experiments and also silent about the impact of heterogeneity on this kind of system. This work has broadened the horizon by exploring different interesting implementation avenues like retail, and showed through a set of experiments that we can find a set of stable of landmarks in spite of the heterogeneity.

IX. Conclusion

We have identified the factors that might affect the stability of landmarks namely device, time and the phone's user. An extensive study has shown us that even though they affect the stability of landmarks, their level of impact is varied. Device heterogeneity, being the major factor (even though the phones are from the same manufacturer's same series). Time heterogeneity exists, which is expected as the surroundings change from time to time. User level heterogeneity is being the least of them. This result is assuring because modeling the heterogenity of device is easiest and one can build separate virtual landmark database pertaining to each class of device. On the other hand, if the result would have varied across users, identifying similar class of users and building database corresponding to each class would have been impossible. The stability of landmarks make *RetailGuide* application more robust and real-world ready. This application, if deployed, will open the new horizon in smart retail analytics.

Acknowledgemnts. We sincerely thank Prof. Romit Roy Choudhury and Saikat Guha for some helpful discussions, as well as the anonymous reviewers for their valuable feedback. This work is financially supported in part by the grant given by Vodafone and ITRA.

References

[1] H. Wang *et al.*, "No need to war-drive: unsupervised indoor localization," in *ACM MobiSys*, New York, NY, USA, 2012.

[2] L. S. Ravindranath *et al.*, "Code In The Air: Simplifying Sensing and Coordination Tasks on Smartphones," in *ACM HotMobile*, La Jolla, CA, USA, 2012.

[3] R. Feliz *et al.*, "Pedestrian tracking using inertial sensors," *Journal of Physical Agents*, vol. 3, no. 1, 2005.

[4] F. Li *et al.*, "A reliable and accurate indoor localization method using phone interial sensors," in *ACM UbiComp*, Pittsburgh, PA, USA, 2012.

[5] M. Zhang and A. A. Sawchuk, "A feature selection-based framework for human activity recognition using wearable multimodal sensors," in *BodyNets*, Beijing, China, 2011.

[6] D. T. Pham *et al.*, "Selecting k in k-means clustering," *Proceedings of IMechE*, 2005.

[7] P. Bradley and U. Fayyad, "Refining initial points for k-means clustering," *International Confernce on Machine Learning*, 1998.

[8] H. Durrant-Whyte and T. Bailey, "Simultaneous localisation and mapping (slam): Part i the essential algorithms," *IEEE Robotics and Automation Magazine*, 2006.

[9] "GitHub Repository of RetailGuide App," https://github.com/ananthbalashankar/landmarkerGSoC.

[10] S. Dey, N. Roy, W. Xu, and S. Nelakuditi, "Leveraging imperfections of sensors for fingerprinting smartphones," in *ACM Hotmobile*, Jekyll Island, Georgia, USA, 2013.

[11] H. G. Walraff, *Avian Navigation: Pigeon Homing as a Paradigm.* Springer, 2005.

[12] C. Buehlmann *et al.*, "Desert ants learn vibration and magnetic landmarks." *PLoS One*, vol. 7, no. 3, 2012.

[13] B. A. Cartwright and T. S. Collett, "Landmark maps for honeybees," in *Biological Cybernetics*, 1987.

[14] D. Kumaran and E. A. Maguire, "The Human Hippocampus: Cognitive Maps or Relational Memory?" *Journal of Neuroscience*, vol. 25, no. 31, 2005.

[15] K. Chintalapudi *et al.*, "Indoor localization without the pain," in *ACM MobiCom*, Chicago, Illinois, USA, 2010.

[16] M. Anisetti and others., "Advance localization of mobile terminal in cellular network," *I. J Communication, Network and System Sciences*, 2008.

[17] M. Azizyan *et al.*, "Surroundsense: mobile phone localization via ambience fingerprinting," in *ACM MobiCom*, Beijing, China, 2009.

[18] C. Xu *et al.*, "Improving rf-based device-free passive localization in cluttered indoor environments through probabilistic classification methods," in *IPSN*, Beijing, China, 2012.

[19] I. Smith *et al.*, "Place lab: Device positioning using radio beacons in the wild," in *Pervasive*, Munich,Germany, 2005.

[20] M. Youssef *et al.*, "Gac: Energy efficient gps-accelerometer-compass-gsm localization," in *IEEE GLOBECOM*, Miami,Florida,USA, 2010.

[21] A. Thiagarajan *et al.*, "Vtrack: accurate, energy-aware road traffic delay estimation using mobile phones," in *ACM SenSys*, Berkeley, California, 2009.

[22] Q. L. Xiaojun Zhu and G. Chen, "Apt : Accurate outdoor pedestrian tracking with smartphones," in *IEEE INFOCOM*, Turin,Italy, 2013.