Incorporating User Motion Information for Indoor Smartphone Positioning in Sparse Wi-Fi Environments

Wasiq Waqar
Department of Computer Science
Memorial University of Newfoundland
St. John's, Canada
wasiq.waqar@mun.ca

Yuanzhu Chen
Department of Computer Science
Memorial University of Newfoundland
St. John's, Canada
yzchen@mun.ca

Andrew Vardy
Department of Computer Science and Department of Electrical & Computer Engineering
Memorial University of Newfoundland
St. John's, Canada
av@mun.ca

ABSTRACT

Indoor localization using mobile devices such as smartphones remains a challenging problem as GPS (Global Positioning System) does not work inside buildings and the accuracy of other localization techniques typically comes at the expense of additional infrastructure or cumbersome war-driving. For such environments, we propose a localization scheme which uses motion information from the smartphone's accelerometer, magnetometer, and gyroscope sensors to detect steps and estimate direction changes. At the same time, we use a Wi-Fi based fingerprinting technique for independent position estimation. These measurements along with an internal representation of the environment are combined using a Bayesian filter. This system will allow us to reduce the amount of training required and work in sparse Wi-Fi environments. We test our approach in two real-world environments to show the benefits of incorporating user motion for indoor localization.

1. INTRODUCTION

In the past, most of the attention was given to Location Based Services (LBS) in outdoor environments as GPS played the dominant role in localization. Recently, we are seeing a paradigm shift in the mobile applications market, where indoor LBS is being considered the new frontier. Due to the increasing number of mega size multi-level constructions like airports, shopping malls, universities and other facilities, people tend to spend more time indoors. Research shows people only spend 10-20% of their time outdoors [1] and more than 70% calls originate from indoors which indicates great potential fot indoor LBS.

The proliferation of smartphones is motivating researchers to look at other ways for more reliable and energy efficient indoor positioning of users which have a reasonable tradeoff between accuracy, reliability, cost, and scalability. To minimize deployment and infrastructure costs, different techniques and technologies are being explored. Indoor positioning is challenging as GPS does not work inside buildings so most common solutions take advantage of existing RF (Radio Frequency) infrastructures like Wi-Fi and cellular. There are several ways in which RF signals can be used for positioning. It is not easy to model the radio propagation in indoor environments because of diffraction, scattering, shading, severe multipath, low probability for availability of line-of-sight (LOS) path, and specific site parameters such as floor layout, moving objects, and numerous reflecting surfaces. There is no single good model for an indoor radio multipath characteristic so far. Different techniques have different advantages and disadvantages. Hence, using more than one type of positioning algorithm at the same time could yield better performance. There are different triangulation, proximity or fingerprinting based algorithms available which deal with the indoor positioning problem in various ways.

On the other hand in robotics, inertial sensors, laser rangefinders, and computer vision are used to provide accurate localization without the requirement of fixed infrastructure. Mobile devices, such as smartphones and music players, have recently begun to incorporate a powerful yet diverse set of sensors. These sensors include GPS receivers, microphones, cameras, proximity sensors, magnetometers, temperature sensors, accelerometers, and gyroscopes. In the future, other sensors like altimeters, barometers, etc., may be incorporated into these devices. Inertial measurement units (IMUs) like accelerometers and gyroscopes are being embedded in most of the latest smartphones. Accelerometers measure 3D linear accelerations of the device whereas gyroscopes give angular velocities. Most modern smartphones also include a magnetometer for raw magnetic readings and heading information. Using these sensors one can estimate the user's motion and characterize their activity as, for example, walking, standing, jumping, running etc. User motion can then also be used to keep track of position via dead reckoning.

Problems arise when using RF based positioning schemes in environments where RF signals are sporadic or sparsely deployed. Due to the placement of APs (Access Points) and cell towers, there might be areas where RF signals are not available. Similarly there may be disruption in the RF sig-

MSWiM'14, September 21–26, 2014, Montreal, QC, Canada.

http://dx.doi.org/10.1145/2641798.2641812.

nals due to limits on radio range, energy resources, and other sources of noise. In such environments it is better to incorporate additional information from IMUs for localization with opportunistic RF based position correction.

Our main contributions to address the above challenges can be summarized as follows:

- **We identify an opportunity to use sensor-based dead-reckoning and opportunistic Wi-Fi positioning for localization using smartphones in areas where there is sparse Wi-Fi coverage.** Our approach does not require the installation of additional infrastructure.
- **We developed and used an iOS app on the Apple iPhone 4 to evaluate our technique. This app was tested in the tunnels of Memorial University of Newfoundland which have very limited Wi-Fi coverage.**

The subsequent sections expand on each of these contributions, beginning with a short related research overview followed by our proposed idea, evaluation, and conclusion.

2. RELATED WORK

Smartphone accelerometers have been used in some mobile localization schemes in an assistive or collaborative manner. In Surroundsense [9], they are used as one of the parameters for the fingerprint, whereas CompAcc [2] uses them to count the number of steps taken to estimate the distance travelled by a pedestrian.

In [11] the authors gave a novel particle filtering based scheme for indoor positioning which does not rely on any infrastruction and uses only the sensors from the smartphones. But their system is not stand alone as their design requires a centralized system. In [6] the authors dont rely on any Wi-Fi but depend on a more accurate step counter and turn detections for position accuracy. We feel that in buildings where multiple floors have the same layout, this scheme might fail and some kind of auto correction measure has to be taken. In other work [10][3][12][5], researchers have used accelerometer data to detect human activities such as walking, standing, climbing stairs, jogging, etc. A short overview of related work is covered in [14].

3. SYSTEM ARCHITECTURE

In probabilistic robotics, a *belief* is the internal knowledge of the robot or a system about the state of the world. In our case state means the location of the subject in our environment. States cannot be measured directly, but we can represent and estimate the probability that the system lies in each possible state. We use the term *belief* to refer to the conditional probability distribution over all possible states. This distribution assigns a probability to each possible hypothesis with regards to the true state. State x_t is generated stochastically from state x_{t-1} meaning that the belief at time t is calculated from its past belief at time $t-1$. The most general algorithm for calculating beliefs is given by the *Bayes filter* algorithm. Algorithm 1 depicts Bayes Filter which is a recursive Bayesian state estimation technique utilized in mobile robotics and other applications [13].

This algorithm is recursively applied at every iteration when belief $bel(x_t)$ needs to be calculated from $bel(x_{t-1})$.

Algorithm 1: *The general algorithm for Bayes filtering*

Input: u_t, z_t, $bel(x_{t-1})$

1: **for all** x_t **do**
2: $\quad \overline{bel}(x_t) = \int p(x_t|u_t, x_{t-1}) bel(x_{t-1}) dx_{t-1}$
3: $\quad bel(x_t) = \eta p(z_t|x_t)\overline{bel}(x_t)$
4: **end for**

Output: $bel(x_t)$

Bayes filter possesses two essential steps. In Line 2, it processes the control u_t. It does so by calculating a belief over the state x_t based on the prior belief over state x_t and the control u_t. u_t in our case is the motion captured from the motion model. This step of the algorithm is also called *prediction* [13].

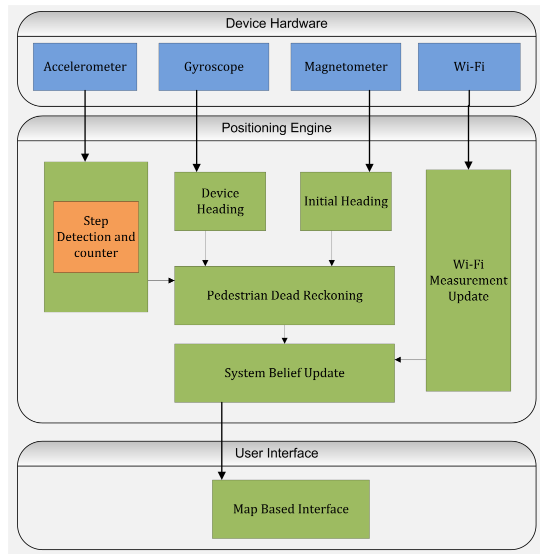

Figure 1: Block diagram of the proposed system

The second step of Bayes filter is called the measurement update. In line 3, the Bayes filter algorithm multiplies the belief $\overline{bel}(x_t)$ by the probability that measurement z_t may have been observed. It does so for each hypothetical posterior state x_t. To compute the posterior belief recursively, the algorithm requires an initial belief $bel(x_0)$ at time $t = 0$. If we are ignorant about the initial condition we can initialize using the uniform distribution.

3.1 Design Overview

Figure 1 shows the block diagram of our proposed system. In our localization scheme we divided our map into a grid. The center of these grid cells are referred to as anchor points which have known physical coordinates (x, y). The grid space between two anchor positions determines the resolution or granularity of the positioning system. The system state variable x_t indicates the anchor point that is closest to the current position. The initial belief of the system is assumed to be uniform as the system will not know where the user is positioned. The on-board magnetometer is noisier compared to gyroscope when giving heading estimation [15]. Therefore, we only use the magnetometer to initialize the orientation of the user and calibrate the gyroscope. This is one of the assumptions of our system that we ask the user to face one of the corridors (potential path where the user can walk). After this initialization/calibration process we keep track of the heading using the gyroscope. We use the step counter [15] to estimate the distance travelled and the gyroscope to estimate the direction in which this distance is travelled. As shown in the figure 1, accelerometers are used to detect the steps taken.

3.2 Motion Model

Using a step counter and gyroscope one can estimate the user's recent trajectory and then predict $\overline{bel}(x_t)$. Step detection is the automatic determination of the moments in time at which footsteps occur. If one wants to use accelerometer data to detect just the instant motion of the device, one needs to be able to isolate sudden changes in the movement from the constant effect of gravity.

Peak detection is a method which calculates the steps from the 3-axis accelerometer readings. A threshold value can be used to detect a peak. If changes in acceleration are too small, the step counter will discard them. The step counter can work well using this algorithm, but sometimes it can be overly sensitive. The algorithm that we chose for our step counter is inspired by an analog pedometer [17]. The algorithm used for our step counter using mobile phone accelerometer is available in [15]. There are several other algorithms available for step counters but most of them are primarily for accelerometers attached to the foot, hip or other body part.

The iPhone 4 has a 3-axis gyroscope which can measure angular velocities about the axes. The Core Motion Framework of the Apple iOS SDK also provide us access to built in functions which manage and keep track of the device's attitude after the application starts. Rotation around the z-axis is called yaw and at the start of the application it is calibrated with the initial stable magnetic heading. The comparison and performance of estimating direction with gyroscope compared with magnetomenter is discussed in [15].

3.2.1 *Belief Update Strategies*

To study our motion model, we divided our map into grid spaces. The centers of these grid spaces are the anchor points which have known physical coordinates (x, y). A set of anchor points is maintained and the probability distribution over this set is represented by $bel(x_t)$. Figures 11 and 12 show the test environments and the positions of all anchor points.

In the time interval $[t-1, t]$ the user advances from position x_{t-1} to position x_t. The step counter and gyroscope report back the relative change in position (x_{rel}, y_{rel}). As we know the initial heading and current heading of the user, we can determine the user's direction of travel. So from the last position and the new position we can determine x_{rel} and y_{rel} which are distances travelled in the x-direction and the $y-$direction with respect to our map.

$$x_{rel} = \alpha \cos(\theta + \beta) \tag{1}$$

$$y_{rel} = \alpha \sin(\theta + \beta) \tag{2}$$

where θ is the initial orientation of the device during initialization, β is the yaw of the device and α is the step length.

The corresponding relative motion parameters (x^*, y^*) for the given poses x_{t-1} and x_t are calculated in lines 1 and 2. These basically come from the known positions in the map. The function $norm(a, b)$ implements an error distribution over a with zero mean and standard deviation of b which was empirically chosen as 4m. The motion model is used as step 2 in our Bayes filter implementation.

3.3 Wi-Fi Fingerprinting

In classic fingerprinting algorithms, vectors of Received Signal Strength (RSS) measured in online phase and offline phase are directly compared to each other. The nearest neighbour method simply calculates the euclidean distance in signal space between the live RSS reading and the fingerprint. A major drawback of using this technique is that different devices, becasue of their hardware and software (sometimes devices of the same make and model), report different RSS values which may differ from the RSS stored in the database. This will degrade the performance of the positioning system. In contrast, rank based localization [8] uses only ranks of the RSS values because the rank information is less sensitive to any bias and scale.

Figure 2 shows the block diagram of the rank based fingerprinting algorithm. In this algorithm, first the RSS values measured in the online phase from different APs are first sorted from strongest to weakest. Ranks $(1, 2, 3, ...)$ are assigned to APs based on the position in the sorted vector. Rank 1 is given to the strongest AP, meaning with the highest RSS value. Rank vectors are created from the fingerprints stored in the database. Ranks are assigned based on the MAC address and rank of AP in the online phase. Then this vector is also sorted strongest to weakest keeping the rank assigned to them. In ideal cases, the sorted ranked vector from online phase and sorted ranked vector fron offline phase will be identical hence showing perfect similarity.

In case an AP which was in the online phase was not found in the database, the rank vector created from the database is padded with 0, to achieve the same length as the rank vector from the online. Other techniques, including via Gaussian kernel [4] which calculates the likelihood of an anchor point using the RSS value similarity between two vectors, also face the dimension mismatch problem. In real indoor environments the dimension of the fingerprints of different anchor points vary considerably. If simple likelihood calculation mechanism (e.g., Euclidean distance or cosine similarity) is used, mismatching could lead to large positioning errors.

Spearman's footrule distance measures total elementwise displacement between two vectors. It is similar to the Manhattan distance for quantitative variables. According to [7] Spearman's footrule perform the best amongst other similarity measures. Assuming u_k is the rank of the k-th element in vector U, v_k is the rank of the k-th element in vector V and n is the number of elements in vectors U and V, Spearman's footrule distance can be computed as follows:

$$D_s = \sum_{k=1}^{n} |u_k - v_k|.$$

The similarity measure above return the scores for every anchor point. The anchor point with the lowest score is considered the best match. Ideally using k smallest reference points to calculate the estimated position yields a better result. In [7] the author solves a p-center problem to estimate the final position estimate. In the rank based technique the

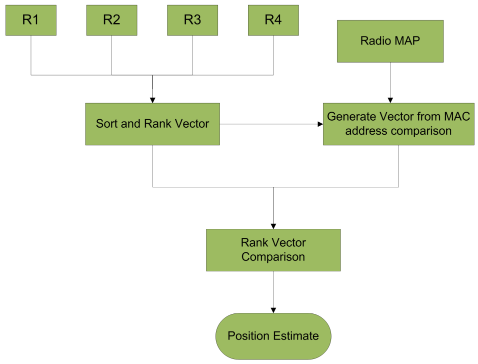

Figure 2: Block Diagram of the Wi-Fi Rank based fingerprinting

distribution of scores will differ because of several reasons. The number of APs visible in the querying scan and position where the scan was done affects the distribution of the scores. For instance if the scan is done at a corner where 20 APs are visible compared to another location where only 5 APs are visible, the distribution of scores will differ a lot. The random test on 13 anchor points in the Engineering Building was done. It was noted that the accuracy of the position estimate was independent from the score distribution. Figure 3 shows the maximum and minimum score distribution and Figure 4 shows the normalized entropy of the score distribution. As the user initiates the application, the belief is uniformly distributed. Entropy is a measure of the uncertainty associated with a random variable and is also referred to as the expected value of the information contained in a message, which in our case is the belief. At position 5 to 9 the accuracy was under 8m whereas 1-4 and 10-13 the error was greater then 8m. The best match at position 6 and 8 were estimated the correct position but both the entropy and min-max distribution does not infer a trend. The distribution of scores tells us that our certainty of our position estimate is not dependent on the score distribution. Hence we used a different approach to use Wi-Fi for position correction. We assign weights $w1, w2$, and $w3$ to the best 3 matched anchor points only if they are all within 2 hop neighbors to each other. Otherwise we ignore the Wi-Fi scan. It means that when the Wi-Fi localization module estimates the best 3 matches, the weights are assigned only if each anchor point is at least 2-anchor point distance to any of the other two. For our experiments we assign 0.4, 0.3 and 0.2 weight to the three best matched positions.

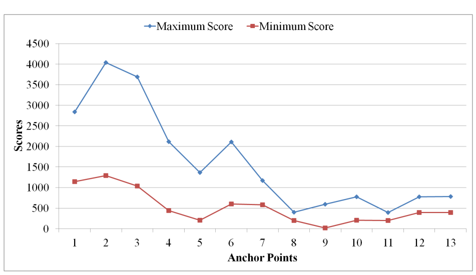

Figure 3: The minimum and maximum scores at different anchor points.

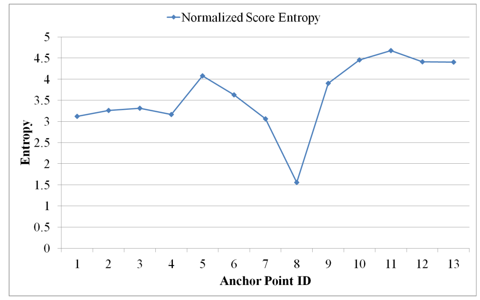

Figure 4: After normalizing the scores, entropy is calculated.

4. PERFORMANCE EVALUATION

We will explain our experiment methodology, settings, scenarios, and results in this section. Our main experimental goal is to measure the benefit of using motion information from the user to track and position in an indoor environment.

4.1 Methodology

The system evaluation contains multiple phases. The first phase is to test the performance of our step counter which is a foundation of our motion model. After checking the accuracy we can determine if it is good enough to be used in our motion model. The accuracy and precision of our motion model is then tested in two different indoor environments.

The second phase is the evaluation of our measurement model. By analyzing the performance metrics, we can determine if it can be used for opportunistic measurement update. Furthermore, it is important to test our system in an environment with sporadic Wi-Fi signal. Next, we explore the benefit of using motion for localization and tracking and analyze the advantages of using rank based Wi-Fi in sparsely distributed Wi-Fi environment. We measure the benefit in the following aspects:

- *System Performance:*

  Hypothesis 1: *The system accuracy and precision is comparable to other Wi-Fi only localization system while using less training*

- *Cost:*

  Hypothesis 2: *The system training and maintenance cost can be reduced.*

- *Scalibility*

  Hypothesis 3:: *The system can work in different indoor environment.*

4.2 Experimental settings

Experiments and evaluations of our motion model, measurement model and hybrid localization scheme were carried out at two contrasting environments at Memorial University. The first area was part of the 2nd floor of Engineering Building. The space was divided into a grid using a 3×3m cell size. 33 positions were selected within the hallways for the anchor points. Each anchor point is surveyed for Wi-Fi data and a fingerprint is created for each survey point. The anchor points are possible locations that the user can be in the environment. The distance between two anchor points is nearly 6 steps. The belief is chosen to be updated after every 6 steps in this environment. Figure 5 shows the map of the Engineering Building field test environment.

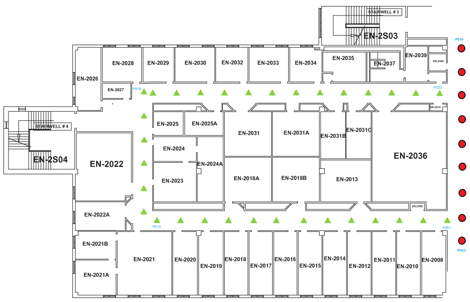

Figure 5: Map of the Engineering Building. Green triangles are the anchor points where data has been collected and the system has fingerprints for those locations. Red circles are untrained areas.

The second environment is the Unversity Tunnel system which connects different buildings of the university. There is no Wi-Fi coverage provided for the tunnels. Figure 6 shows the map of the tunnel system. The only Wi-Fi signals available are at entrance positions to the tunnel. Hence the areas of Wi-Fi AP visibility is very limited and also sporadic in nature. This place is a good testbed for our system. Both the environments are different. The Engineering Building has more sharp turns, whereas the the tunnel has smaller turns. The distance between two anchor points here is 5.5m. Therefore the belief update happens after 9 steps.

The major assumptions for our experiments are as follows

- The user is always located in the areas for which the anchor points are defined in the system.
- The device is always pointing in the direction of the user motion.
- The user walks close to the corridor's center.

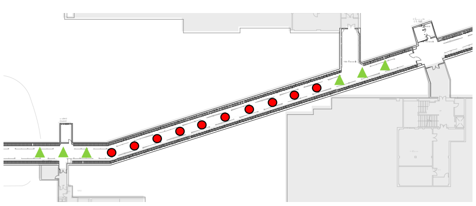

Figure 6: Map of part of the university tunnel. Green triangles are the points where Wi-Fi is sporadically available and red discs are positions where no Wi-Fi is available. Fingerprints for locations with green triangles are available.

4.3 Motion Model Evaluation

In an experiment the user was asked to walk in the corridor with our localization app in the trained areas of Engineering Building. Figure 7 shows the heat map of the probability distribution over time. The x-axis describe the ith update of belief. The position IDs are listed on y-axis where the color intensity shows the probability of being at each location. The belief at x36, x64 and x88 are examples where the position correction happens due to turning. Overall it can be seen that the position is tracked pretty well along the path of the user. From belief update x112 to x128 the user changed his direction of walking after a few steps a couple of times creating a to-and-from user trail. It can be observed in the heat map that the uncertainty starts to increase as the probability distribution spreads out. So a malicious behavior by the user in terms of walking in circles and moving to and fro in the corridor over short distance might confuse the belief system.

Figure 8 shows the entropy of the same heat map. At x5 the entropy falls greatly due to a turn. Initially as the probability was uniform so the entropy was maximum but as soon the user turned the belief became more certain due to the recognition of a corner. Every time the user turns the corner, the uncertainty decreases and we can see a drop in entropy. After x112 the entropy increases a little bit showing the confusion caused by user motion.

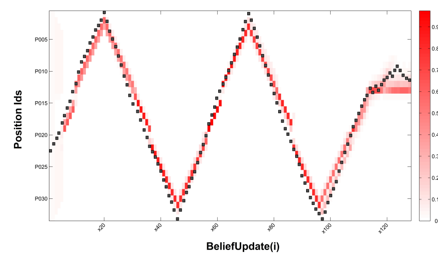

Figure 7: Motion model heat map at Engineering Building with dense Wi-Fi coverage. Black annotations describing actual user position.

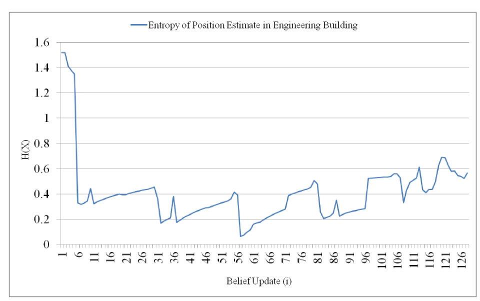

Figure 8: Entropy in the Engineering Building

4.4 Rank Based Wi-Fi Measurement Model Evaluation

Our Wi-Fi localization scheme returns similarity scores between the current measurement and every anchor point which has been surveyed for stored Wi-Fi data. The lowest score is considered the best match. To test the rank based fingerprinting technique we assumed that the best match anchor point is the estimated position. We tested this in our Engineering Building where we tested it at each anchor point. The error was recorded by logging the distance between the ground truth and the estimated output position. Figure 9 shows the cumulative error distribution. The mean error was about 4.1m. We compared our system with the Wi-Fi based localization scheme by Yan et al [7] which uses a completely different approach for localization.

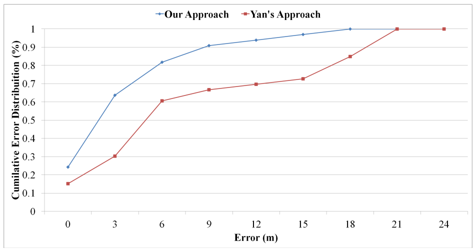

Figure 9: Cumulative error distribution of the Rank Based fingerprinting in Engineering Building

4.5 Performance in Sparse Wi-Fi Environment

To test our system in an environment which has sporadic Wi-Fi signals we chose the university tunnel system which has no Wi-Fi available but sporadic signals are available at the different entrances of the tunnels from different buildings. Figure 6 shows the map of one such section of the tunnel. This figure shows 16 anchor points from one entrance to another. All anchor points are equally distant from each other. It is assumed that initially the system does not know the user's true position. Initializing with a Wi-Fi scan can initialize user position if the user is in one of the entrance areas.

Figure 10 shows the heat map of the user's walk in the tunnel. On x-axis we have the belief updates and on y-axis we have the 16 anchor points. We annotated the map with approximate actual position of the user to compare the belief distribution with the movement of the user. From x0 to x12 we can see that the belief is randomly distributed but it converges towards one direction. From x12 to x45 the probability distribution is not that scattered and position estimates are more confident. From x45 to x60 the probability distribution becomes less reliable as the user changes his direction more frequently similar to the test done in Engineering Building. At x60 the Wi-Fi measurement update is triggered. At this point it detects P001 as the most likely position. The probability distribution shifts heavily towards that position as we give higher weight to the anchor points with higher Wi-Fi similarity. In the tunnels the Wi-Fi is sporadically available in only P001-P004 and then P015-P016 as described before. No Wi-Fi is detected in any anchor points between them. Hence when the Wi-Fi update step is triggered, due to the diversity of visible AP's between these two regions, the position correction has smaller error.

Figure 11 shows the entropy of the belief in the tunnel. If we compare the entropy graphs of Engineering and tunnel it can be observed that the entropy in the tunnel does not drop as much as compared to the entropy in the Engineering. This is because the tunnel lacks sharp turns as compared to the Engineering Building. Although the accuracy from the most probable position estimate is comparable in both locations the certainty is less because of the absence of sharp turns. At x51 to x59 it can be observed that due to the frequent turning around in the same corridor the entropy increases. It sharply decreases again at x60 when Wi-Fi measurement update is triggered.

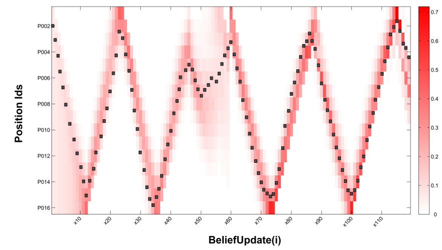

Figure 10: Heat map of motion model in the tunnel with sparse Wi-Fi

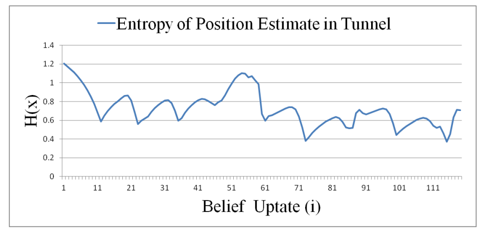

Figure 11: Entropy in the Tunnel

5. CONCLUSION

This paper explores the idea to incorporate user motion for indoor localization for sparse Wi-Fi environments, where other Wi-Fi only systems would not perform accurately. We discuss our results in relation to the three hypotheses mentioned earlier.

- *System Performance*

  Hypothesis 1: *The system accuracy and precision is comparable to other Wi-Fi only localization system.* As it can see from the heat maps of both environments that the system tracks the user. For our experiments the accuracy at Engineering Building was under 4m whereas in the tunnels it was around 6m on average. It is marginally worse than the 0.7m to 4m average positioning error yielded by the best-performing but intensively trained Horus system (using 100 Wi-Fi scans and much smaller grid space (1.52 m and 2.13 m)) [16]. But generally for our system a single accuracy figure can not be given as it depends upon the shape and size of the environment. Sharp turns help reduce positioning error estimates and long corridors accumulate errors. The second factor is the amount of Wi-Fi landmarks available for position correction.

- *Cost*

  Hypothesis 2: *The system training and maintenance cost can be reduced.* We tested our system in two different environment. One which had very dense Wi-Fi and had training data available for all the anchor points. On the other hand in the tunnel environment the Wi-Fi was sporadically available at only 6 locations. There was no survey done for those anchor points. As different areas in such environment have distinct Wi-Fi visibility, so it can be exploited to our advantage to correct the position only and rely more on human motion for positioning. Due to less reliance on Wi-Fi, minor changes in Wi-Fi infrastructure would not affect the system.

- *Scalability*

  Hypothesis 3: *The system can work in different indoor environment*. We tested our system in two completely contrasting environments. One had sharper turns with denser Wi-Fi coverage and the other having few turns but sparse Wi-Fi coverage. The grid size in both the environment was also different. This system is more scalable than other indoor positioning systems as it would require less training and would even work in sporadic Wi-Fi environment where Wi-Fi only systems would fail.

6. FUTURE WORK

We believe that this system can be further improved in a lot of ways. For example in the step counter we are detecting the number of steps taken but using the height of the user as a parameter to determine the stride length. Perhaps more adaptive approach can be used here which uses information from accelerometer to also calculate the stride length. Artificial intelligence techniques can be employed in the initialization phase for the system to learn the human walking pattern and determine the style of the user to more accurately determine the number if steps.

Similarly for Wi-Fi based localization, preprocessing the APs after observing the environment for fluctuations can be done which might improve the localization error.

Another interesting aspect in which the system can be improved is to integrate human-centric collaborative feedback. Positioning accuracy and precision can be improved by collecting both positive and negative feedback from users in terms of orientation.

Developing a magnetic map is also one idea which can be explored. In that case we have to observe how stable is the magnetic environment over time. In indoor environments there may be areas due to electronic equipment or wiring, where the magnetic field perturbations are distinctive. They can be used as landmarks similar to how we use Wi-Fi.

We believe that some organizations or companies will devise specifications for indoor positioning system in the near future. With the potential rapid growth of location-aware services for public indoor environments such as airports, subway systems, museums, university campuses, shopping centers, etc there will always be areas where Wi-Fi infrastructure will not be available and hence some alternative technology would be needed which is reliable and scalable at the same time. At this time we believe human motion based localization schemes have great potential and look to be very promising in reducing the cost both in the sense of maintenance and energy consumption. We also believe that more and more researchers will be attracted to exploit the various sensors now available in smartphones for indoor localization.

7. REFERENCES

[1] Indoor LBS And Hyper - Local Content Is the Next Gold Rush for Mobile Commerce. http://www.indoorlbs.com/2010/06/indoor-lbs-and-hyper-local-content-is.html, October 2010.

[2] Ionut Constandache, Xuan Bao, Martin Azizyan, and Romit Roy Choudhury. Towards Mobile Phone Localization without War-driving. In *Proceedings of the 29th Conference on Information Communications (Infocom)*, pages 2321–2329, 2010.

[3] Adil Mehmood Khan, Young-Koo Lee, Sungyoung Lee, and Tae-Seoung Kim. Human Activity Recognition via an Accelerometer-Enabled-Smartphone Using Kernel Discriminant Analysis. In *Proceedings of the 5th International Conference on Future Information Technology*, 5 2010.

[4] Azadeh Kushki, Konstantinos N. Plataniotis, and Anastasios N. Venetsanopoulos. Kernel-based positioning in wireless local area networks. *IEEE Transactions on Mobile Computing*, 6(6):689 –705, 6 2007.

[5] Jennifer R. Kwapisz and Gary M. Weissand Samuel A. Moor. Activity Recognition using Cell Phone Accelerometers. In *Proceedings of the Fourth International Workshop on Knowledge Discovery from Sensor Data*, 2010.

[6] Fan Li, Chunshui Zhao, Guanzhong Ding, Jian Gong, Chenxing Liu, and Feng Zhao. A reliable and accurate indoor localization method using phone inertial sensors. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pages 421–430, New York, NY, USA, 2012. ACM.

[7] Yan Luo, Y.P. Chen, and O. Hoeber. Wi-fi-based indoor positioning using human-centric collaborative feedback. In *Communications (ICC), 2011 IEEE International Conference on*, 6 2011.

[8] J. Machaj, P. Brida, and R. Piche. Rank based fingerprinting algorithm for indoor positioning. In *Indoor Positioning and Indoor Navigation (IPIN)*, sept. 2011.

[9] Azizyan Martin, Constandache Ionut, and Roy Choudhury Romit. SurroundSense: Mobile Phone Localization via Ambience Fingerprinting. In *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 261–272, 2009.

[10] Andrew Offstad, Emmett Nicholas, Rick Szcodronski, and Romit Roy Choudhury. AAMPL: Accelerometer Augmented Mobile Phone Localization. In *Proceedings of the 1st ACM international workshop on mobile entity localization and tracking in GPS-less environments*, pages 13–18, New York, NY, USA, 2008. ACM.

[11] Anshul Rai, Krishna Kant Chintalapudi, Venkata N. Padmanabhan, and Rijurekha Sen. Zee: Zero-Effort Crowdsourcing for Indoor Localization. In *Proceedings of the 18th annual international conference on Mobile computing and networking*, Mobicom '12, pages 293–304, New York, NY, USA, 2012. ACM.

[12] Nishkam Ravi, Nikhil Dandekar, Prreetham Mysore, and Michael L. Littman. Activity recognition from accelerometer data. 2005.

[13] Sebastian Thrun. Probabilistic Robotics. *Communications of the ACM*, 45(3):52–57, March 2002.

[14] Wasiq Waqar, Yuanzhu Chen, and Andrew Vardy. Exploiting smartphone sensors for indoor positioning: A survey. In *Proceedings of the Newfoundland Conference on Electrical and Computer Engineering*, 2011.

[15] Wasiq Waqar, Andrew Vardy, and Yuanzhu Chen. Motion Modelling using Smartphones for Indoor Mobilephone Positioning. In *Proceedings of the Newfoundland Conference on Electrical and Computer Engineering*, 2011.

[16] Moustafa Youssef and Ashok Agrawala. The Horus WLAN location determination system. In *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*, pages 205–218, 2005.

[17] Neil Zhao. Full-Featured Pedometer Design Realized with 3-Axis Digital Accelerometer. http://www.analog.com/library/analogdialogue/archives/44-06/pedometer.html, 10 2011.