IoTSSC: PARTICLE FILTER BASED INDOOR LOCALIZATION WITH BLE AND CLOUD

Ezenwafor Chudalu, s1777982

ACM Reference format:
Ezenwafor Chudalu, s1777982. 2016. IoTSSC: PARTICLE FILTER BASED INDOOR LOCALIZATION WITH BLE AND CLOUD. In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 8 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Positioning and localization has been a topic that has gained in popularity [3]. Multiple concepts of computing such as ubiquitous computing and Internet of Things, location or context-aware systems are dependent on positioning or localization [3]. The importance of localization has risen a lot due to its applications in various industries. Applications can be seen in military, space exploration, telecommunication, Internet of Things (IoT), Robotics and several other fields. The development of various industries utilizing the information of individual location, and the fast growing rate of users of smart-phones has accelerated the development of positioning and localization techniques used globally [5] and locally in indoors. The development and usage of Global Navigation Satellite System (GNSS) has successfully achieved good accuracy and proper localization of individuals, buildings or any desired object. However the GNSS has been very limited in indoor environments and the position accuracy can be poor in tightly-packed urban areas with tall buildings. various methods and techniques has been proposed over the years, such as: Wireless Local Area Network(WLAN), Bluetooth, Ultra-wideband (UWB), Visible Light Communication (VLC), and Pedestrian Dead Reckoning (PDR) etc [4]. WLAN have been used for indoor positioning in various places, with several methods such as radio signal strength with wireless signal propagation model, triangulation and fingerprinting. Most of these model requires a lot of resources to implement indoor localization systems. Hence, in the field of IoT, several methods and techniques have been implemented with the aid of off-the-shelf sensors and affordable components, which has allowed the development of various localization techniques. These localization techniques revolves mostly around the techniques listed below:

- Pedestrian Dead Reckoning (PDR) [6, 10],
- Utilization of BLE Beacons and Triangulation [3],
- Map Matching [5],
- Fingerprinting [9],
- Hybrid (combination of multiple methods) [5, 7]

Conference'17, Washington, DC, USA

DOI: 10.1145/nnnnnnn.nnnnnnn

The implementation of indoor localization has been worked on over the years using several methods, as listed earlier. This report paper presents an IoT solution of a Particle Filter based indoor localization system. The localization problem is tackled as a coursework project of Internet of Things:Systems, Security and Cloud course. The indoor localization problem required the indoor position estimation of an individual in several lab rooms of the 5th floor of Appleton Tower of the University of Edinburgh. To solve this problem, various recourses were provided and some used, as seen below:

- Sensor: Adafruit LSM9DS1
- BLE Beacons
- Motorola Android Smartphone
- Custom cloud

The beacons were deployed in the lab halls and corridor. Using this and the other listed resources above, it was required to develop and design a localization solution, that also captures the essence of IoT. It was then required to define a product or application using the proposed solution. This report paper introduces and presents the undertaken solution developed to tackle the problem of indoor localization. Using the resources provided, the proposed solution involves:

- A set of processes hosted on the Nordic NRF51-DK board, such as a step tracker with a sliding window that uses input from the accelerometer, a step distance estimation invoked by the step tracker, a heading estimation using input from the magnetometer and the BLE beacon scanner with a Beacon table/registry. These processes enable the required inputs to be processed accordingly. Hence, a packet consisting of the distance moved, heading direction, and beacons sensed, is sent over bluetooth connection to the android phone.
- The android phone hosts a set of processes, such as a particle filter that takes in the packet received, and uses the distance moved and heading estimate as its input for the motion model, and the beacon sensed is taken as an input for its sensor model, which is designed using an occupancy grid map of the floor plan, and beacon reception maps of each of the beacons. These information are processed to give an estimate of the user's location, and then an HTTP Post process is involved, which sends the estimate to the cloud.
- The cloud host a set of services, where the estimate is received and stored in the database, and the estimates are used as inputs for visualization of the user's estimated position, Motion tracking, and Heat map. The cloud can also output its dataset as a .csv file for further use or analysis.

The description given above would be treated in much details in the chapters and sections ahead. The organization of this report paper is as follows:

(1) INTRODUCTION: Introduced the concept and problem of Indoor localization, and brief description of solution.
(2) REVIEWED WORKS: Presents brief mentions and discussions of some related works.
(3) METHODOLOGY: Introduces the approach and design used, with detailed discussion and description of Hardware (sensors and board) and software (android application) and cloud implementation(web page, visualization).
(4) EXPERIMENTS, DATA ANALYSIS AND PROCESSING: presents techniques used to process input data and experiments done.
(5) RESULTS AND ANALYSIS: Presents the results of the designed solution, and its evaluation.
(6) CONCLUSION: discussions on future works and improvements that can be made are treated here, with a general discussion on the solution and its localization performance.

2 REVIEWED WORKS

The problem of Indoor localization has been handled using various approaches over the years. In the Thesis work [3], they designed an indoor localization solution that implements a step detection function, Particle filter estimation and a optional Map-based approach. The solution [3] uses data from sensors found on modern, off-the-shelf smart phones. it uses the map of the premises and calculates the position by using a particle filter. The processing of the sensor information also included the accounting of sensor noise and step detection. The used sensors are accelerometer, magnetometer and Bluetooth Low Energy (BLE) chip. The BLE chip was used to scan for beacons and deliver the Received Signal Strength (RSS) values, which is combined with the Universally Unique Identifiers (UUID) from all beacons in range. the RSS values are converted to distance measurement. The distance measurements are then used to update and weight the belief in their Particle Filter. The estimated user position is derived from the particle filter. In [5], an indoor positioning system with Inertial Measurement Unit (IMU), Map Matching and Particle filter is designed. A novel position estimation scheme exploiting a smart phone with the IMU sensors is proposed. In this scheme, step detection and step distance estimation and orientation are estimated by using the inertial sensors of a smart phone. Map matching and particle filter techniques are applied to improve performance of positioning. In [10], an inertial navigation system using the limited quality data provided by the inertial sensors in smart phones is built. A probabilistic approach for orientation and use-case free inertial odometry, which is based on double-integrating rotated acceleration is presented. This approach fuses noisy sensor data and learning the model parameters online. The information fusion is completed with altitude correction from barometric pressure readings (if available), zero velocity updates (if the phone remain stationary), and pseudo-updates limiting the momentary speed. The approach is demonstrated using an iPad and iPhone in several indoor dead-reckoning applications and in a measurement tool setup. A review on several popular indoor localization techniques are made in [4, 11].

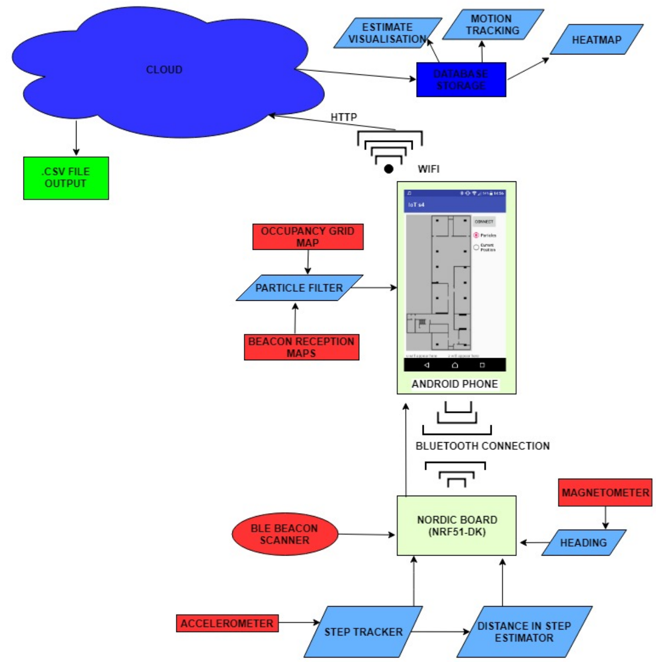

Figure 1: Block Diagram showing the general overview of the implemented IoT indoor localization solution.

3 METHODOLOGY

The concept of indoor localization has been discussed in chapter 1. The description of the proposed and implemented solution to the specified problem of indoor localization presented earlier is handled in this chapter. The organization and implementation of the methods and techniques employed are detailed and discussed. A set of beacons are deployed in the 5th floor of the Appleton building, and some resources as listed earlier, are provided to design and develop an IoT solution for indoor localization. Figure 1 represents the overview of the proposed and implemented solution. The description of the of the localization solution is given in the sections ahead. The description of the methods is divided into Hardware, Software and Cloud implementation, with their following sub-sections.

3.1 Hardware Implementation

The hardware resources used for the hardware (Nordic board) implementation of this solution are described below:

- Nordic NRF51-DK: [8] the nRF51-DK development kit is a single-board development kit for Bluetooth Smart, ANT and 2.4GHz proprietary applications using the nRF51 series System on Chip. This development kit supports ARM Mbed tool-chain for rapid prototyping and development using Mbed's cloud-based IDE and tool-chain with an extensive range of open-source software libraries. Some significant features of this board are: Nordic nRF51 System-on-Chip combining Bluetooth v4.1 compliant 2.4GHz multi-protocol radio, and ARM Cortex-M0 processor on a single chip optimized for ultra-low power operation.

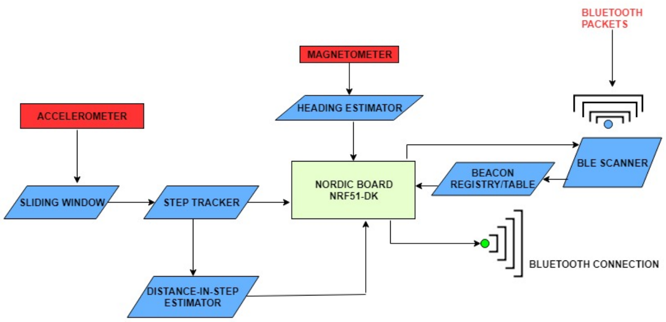

Figure 2: Block Diagram showing the general overview processes handled in the Nordic board.

- Adafruit LSM9DS1: [1] all-in-one 9-DOF sensor that consists of three sensors. A 3-axis accelerometer, which can tell which direction is down towards the Earth (by measuring gravity) or how fast the board is accelerating in 3D space. A 3-axis magnetometer that can sense where the strongest magnetic force is coming from, generally used to detect magnetic north. And a 3-axis gyroscope that can measure spin and twist.
- BLE Beacons: [2] wireless device that periodically broadcast a Bluetooth Low Energy advertising packet, that can be received by a smart phone and used to determine the position with respect to the beacon itself. This allows to provide "context-aware" information to the mobile user, opening up a lot of possibilities.

The components described above are hardware components used to process the required inputs for the hardware implementation of the indoor localization solution, as can be seen in Figure 2. The sections below describes the processes hosted in the Nordic board and the relationship between them.

3.1.1 Firmware Description. The firmware hosted in the Nordic board is responsible for processing the sensor inputs to be used for position estimation and forward these information over bluetooth connection to the android phone. To enable these functionalities, a set of services/processes were hosted, as seen in Figure 2 and 3. The description below briefly discusses the processes in the Nordic board and their communication with each other to deliver a the processed information to the android phone.

- Sliding Window: This is a process that uses an array of defined length (like 25) to receive sensor inputs and pop them out as new inputs are received. In each round (the old input at end of array is popped out, with new entrant), the max input value is found, and 90 percent of the it, is used as the new threshold. This enables adjustment of step detection threshold while user changes or walking style changes. The inputs are root mean square calculations of the accelerometer value in the X, Y and Z-axis.

  $$A_r = \sqrt{A_x^2 + A_y^2 + A_z^2}$$

  The threshold is set and then used for step detection. Two sliding window processes are designed and used for the upper and lower limit adaptation.

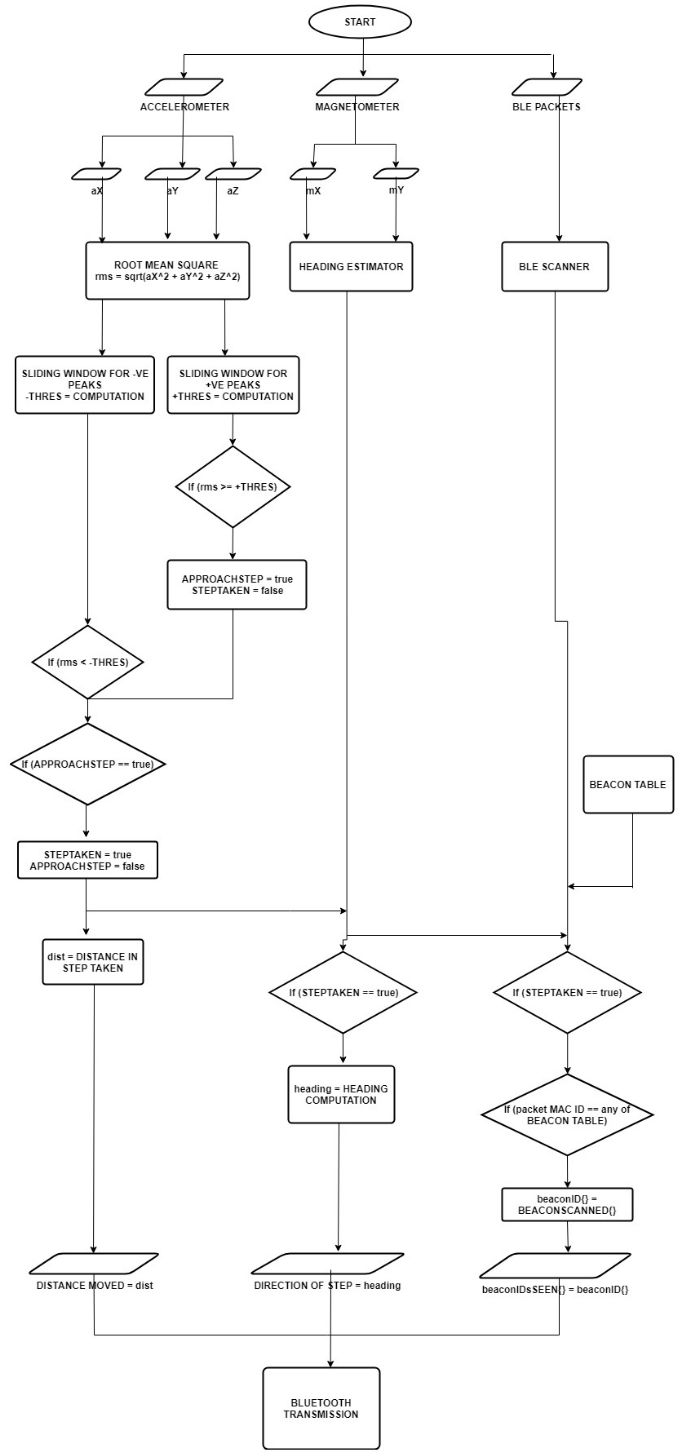

Figure 3: Flow Chart showing processes handled in the Nordic board and their communication.

- Step Detection: This receives input of the accelerometer, which is processed using the root mean square calculation, as seen in Figure 3. The set threshold from the sliding window is used to detect the step, as seen in Figure 3. When a step is detected, this process calls the Distance in step estimation, and the Heading estimation.

- Distance in Step Estimation: This uses the equation of motion to compute the distance moved from when the previous step was taken, to the very recent/current step.
  $Distance = Velocity * time + \frac{1}{2} * acceleration * time$
  $Velocity = velocity_0 + acceleration * time$
  Where the values of acceleration is derived from the accelerometer and the time in between step is estimated with a timer. Due to the inaccuracy of this implementation as a result of noise and drifts in the sensor, a Distance averaging process is called when the distance is calculated, so as to estimate the a better distance moved in relation to the previous 4 distance values, or first 4 distance values when a walk just begins.
- Heading Estimator: This receives input from the X and Y-axis magnetometer reading. After several experimentation and calibration, this process was able to provide good estimation of direction when on flat surface. But this was badly affected by tilt motions. The estimated heading is retrieved every time a step is taken to determine the direction of the motion.
- BLE Scanner: This is a rigorously designed process, that facilitates scans for advertisement packet of beacons deployed in the area. This process makes use of a Beacon table containing the MAC IDs of all deployed beacons. All advertisement packet are scanned for, and anyone that is found is verified by comparing the MAC IDs in the Beacon table with the MAC IDs on the advertising packets. During a walk, multiple beacons packets can be picked up in step intervals. These packets are verified and used to deduce information into an array, which tells the identification of the beacons in the vicinity. Array of length 10 is used which represents the ten beacons deployed, where if any array location ranging from 0 to 9 has a value of 1, then the beacon with that location was sensed.
- Bluetooth Communication: The information gathered during each step interval is converted to simple bytes interpretations that is then easily sent over the bluetooth connection to the android phone by using the Bluetooth GATT characteristics. This packet would be received at the phone end and because the information is place in the GATT Characteristics, the Notifications option is used. This allows continuous sending from the Nordic board, and reception of information as notification to the phone. The (Byte) information encoding is interpreted at the android phone end for further processing. The introduction of new information is triggered by the step detection, and in a split second, these information are encoded and sent. Each step triggers the introduction of new information, and the sending of the current information collected.

The second stage of the process of this localization solution is handled in the android phone, which hosts a set of services and the particle filter. This is discussed in the sections ahead.

3.2 Software Implementation

For the software implementation, an android smart phone was used, which received sensor inputs and are applied as inputs to the motion model and sensor model of the particle filter to provide position estimation. The phone also act as a gateway for the Nordic board to communicate with the cloud. Figure 4 shows the simple services and processes hosted in the android phone. These processes are discussed and some other services employed are also touched on. An android application was developed to host all the required processes described above and in the sections ahead, and services like estimation visualization, gateway for the hardware implementation etc were also implemented.

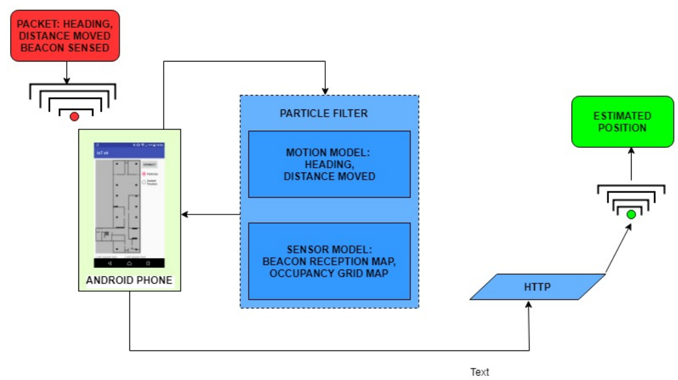

Figure 4: Block diagram showing processes handled in the android phone by using the designed application.

3.2.1 Bluetooth Connection (Packet Reception): The android application developed, facilitated the initiation of a bluetooth connection with the Nordic board, which continously advertises until the connection is made. Once the advertising packet of the board is found, the phone connects to the board by using its MAC ID, which enables swift connection, with no agreements or transactions made. Then the GATT Characteristics are updated from the board, which notifies the android, and allows for the processing of new information. The motion information which comprises of the distance moved and Heading, and the sensor information which comprise of the IDs of Beacon that were sensed during the step interval are derived from the bluetooth transmission and inputed to the particle filter. This can be seen in Figure 4 and 5.

3.2.2 Motion Model: The motion model applies motion to all the particles in their various hypothesis. The heading and the distance in steps are used to determine the direction and distance to move the particles. The particle filter also incorporated some values that determine and represented the variance and noise that can be experienced when a step and direction is estimated.

3.2.3 Beacon Reception Maps: These are maps that were developed rigorously to estimate the reception of the different beacons on every grid location of the area used for the localization problem. They are used in the sensor model to weight the particles by checking the maps to validate their hypothsis. This map also account

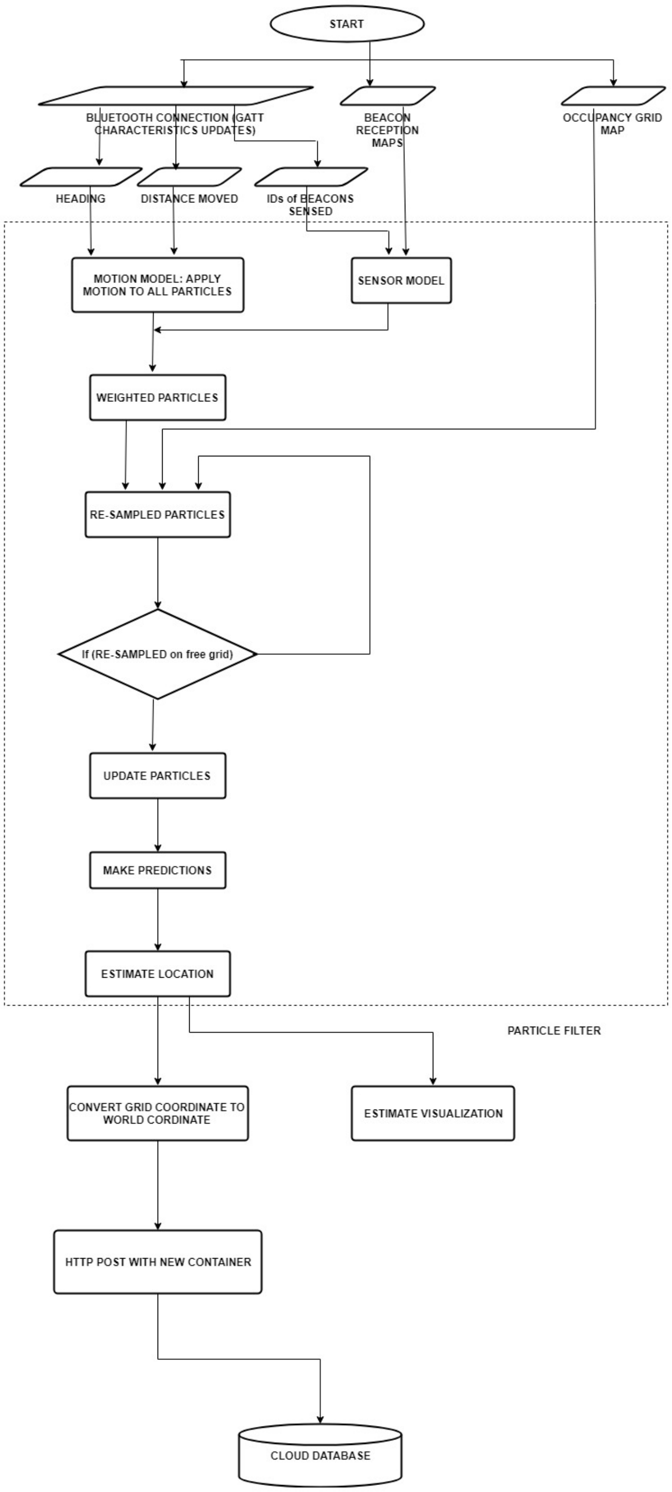

Figure 5: Simple flow chart showing processes handled in the android phone by using the designed application.

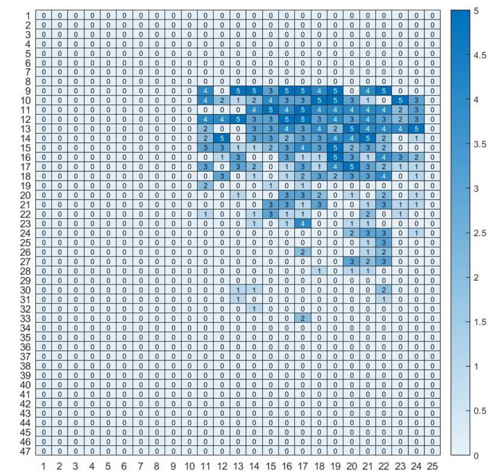

Figure 6: Beacon Reception Map of beacon 9 in 5th floor, Appleton tower

for noise and probability of receiving beacon packets at different locations. Figure 6 shows a beacon map of beacon 9.

3.2.4 Occupancy Grid Map: This is a map that was processed from the picture of the floor plan. The floor plan picture has an estimate of 43.1 pixels to 1 meter ratio. Matlab was used to carve out the grids and overlay it on the floor plan, hence deriving the grid map. This was also constructed digitally using binary matrix. It is used in the re-sampling stage of the particle filter, to make sure particles all lie on free space and not occupied ones. Hence, improving the various hypothesis altogether.

3.2.5 Sensor Model: This model made use of the beacon reception maps, as introduced earlier. The beacon sensed during user motion helps in determining user location or radius. hence by referring to the reception maps, and finding out how likely a beacon is to be sensed from current position, allows the particles to be weighted appropriately. This allows particles with much more likelihood as defined in the reception maps to be weighted higher. The particles are then re-sampled, and particles with low weights die off, and are sampled on particles with higher weight. To enable proper working and better accuracy of the filter, the occupancy grid map is referred to when each particle is re-sampled, to make sure that they are all placed at grids with free spaces. The particles are resampled, which shows that they are updated, and eventually more concrete hypothesis would be found when weighting and re-sampling continues.

3.2.6 Estimate Location and Visualization: The updated particles would likely converge overtime, as various hypotheses gets invalid and particles get re-sampled. The estimated location is then determined by the highest number of particle present in any grid of the grid map. This location is visualized in the phone, where the user can see the particles or the estimated location.

3.2.7 Coordinate conversion, HTTP Post and Cloud Storage: The estimated position is represented in grid position. There is a process defined to convert the grid position into longitude and latitude. The time, grid location and coordinate location of the user is sent as JSON over HTTP Post to the cloud database, by creating a new container.

3.3 Cloud Implementation

The estimated positions derived in the android processes are stored in the database of the cloud. Using a web page in the cloud resource, a simple visualization of the user's estimated location, Motion tracking and Heat map is done. This visualization updates in real time, with less than 2 to 3 seconds, meaning that for each raw information in the board to be processed to an estimated position in the android phone and then visualized in the cloud, it takes less than 2-3 seconds. The option of getting the hard copy of the database is provided, where the customers or users can get it as .csv file. This option can lead to further analysis if needed.

4 EXPERIMENTS, DATA ANALYSIS AND PROCESSING

A lot of experimentation and input data analysis was conducted in early stages of this project. The sections below describe briefly some experiments, analysis and processing that were conducted intensively to obtain this viable solution.

4.1 Ground Truth Measurement of Floor plan

The ground truth measurement of the floor plan could have been tackled in different ways. Instead, given a good quality picture of the floor plan, where a meter was represented by 43.1 pixels, made measurement collection much easier. using Matlab code, the picture was divided into fine grids of 1 meters space. This allowed easy formulation of estimation algorithms used, as explained in the software implementation sections in chapter 3. Figure 7 show the grid map derived from Figure 8.

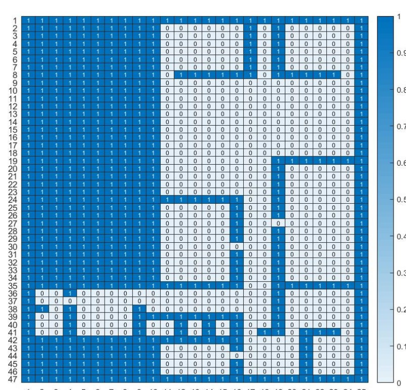

Figure 7: Occupancy Grid Map of the 5th floor of Appleton Tower

4.2 Sensor Input Analysis and Processing

- Accelerometer Reading: For the step tracker to work accurately, the sensor must be able to keep track of the steps in the different types of gaits it might comes across. The input to the step tracker as seen in Figure 9, shows how noisy the sensor reading are. To cope with this, a sliding window process was used to track the steps of the user despite the interferences. Hence a good step tracking solution was developed.
- Magnetometer Reading: In the process of magnetometer input collection and calibration, it was discovered that the it had to compensate for 'Hard iron' and 'soft iron' effects. Figure 10 shows this scenario. Due to the findings, re-calibration was done, and the heading was much more accurate on the horizontal surface. The accuracy was highly influenced with noise by the tilting movements of users. This allowed the use of a heading estimation process that made satisfactory estimations but with low accuracy in some occasions.
- Received Signal Strength (RSS) of Beacons: The RSS values of the beacons deployed in the 5th floor of Appleton tower suffered from very poor signal strength and high variation, with no correlation of the RSS with distance. This introduced difficulty in using the beacons for localization, hence the introduction of beacon reception maps. As one can see in Figure 11, the RSS values do not show any meaning when distance is increased or decreased. Distance can not be deduced from the RSS.

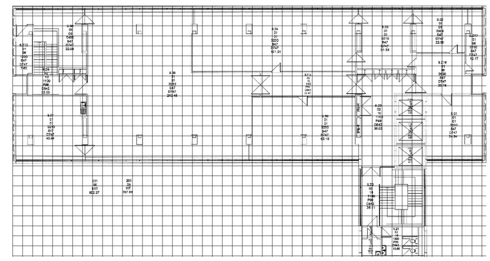

Figure 8: Grid Map of the 5th floor of Appleton Tower

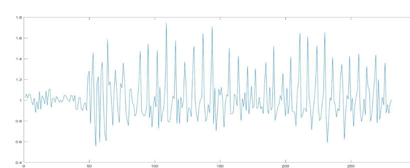

Figure 9: Accelerometer input after Root Mean Square computation

5 RESULTS AND ANALYSIS

The proposed and implemented solution of the Indoor localization problem described for the IoTSSC coursework successfully deployed an IoT solution that could estimate the user's location using off-the-shelf sensor components and various resources. The estimation of the users location was visualized in both the phone and the cloud. After several tests and analysis of the solution and its capability, several evaluations have been made to describe the

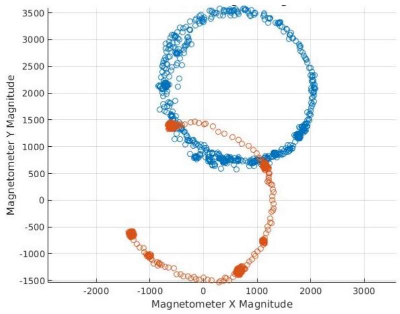

Figure 10: Magnetometer reading before calibration (blue), and after calibration (red)

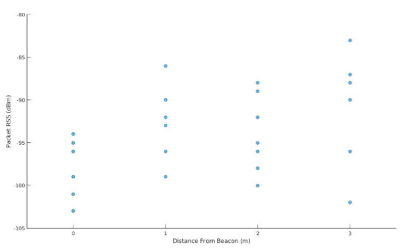

Figure 11: RSS values of a beacon as distance increases

strength and weakness of this solution. The figure shows how the system operates when a walk/run is underway. The user is able to see the real-time visualization of his/her position estimates. The cloud user or customer is also able to see the real time visualization of the user's position estimate, Motion Tracking, which takes into account the previous locations the user was estimated to be in, and the Heat Map, which provides information on what area or trajectory is used more or has been used/visited the most. This is shown in Figure 12.

Figure 12: Visualization seen in implemented solution when a walk/run is underway

5.1 Evaluation

This solution has been evaluated based on four components:

- Accuracy: After several test runs and walks, it was discovered that the step tracker could detect a step and give the number of steps taken with an accuracy of about 95 percent. The general accuracy of the system was found to be variable, such that; sometimes it gives pin-point accuracy of the user's location, or 3 to 5 meters range of accuracy in the estimation, and sometimes the estimations are not good enough. Experimentation showed that, When the particles are evenly distributed, despite the location, and a user walks for about 10 steps, the solution accurately detects the room or hall the user is located in.
- Energy Consumption: This solution lacks good energy efficiency, because most of the processes are handled in the Nordic board and android, which are quite intense. The unreliability of the cloud highly contributed in this decision pushing workload to the phone.
- Latency: This solution takes less than 2 to 3 seconds for each information/process from the board, through the phone and up to the cloud to be fully visualized and processed. Hence this implementation is quite fast.
- Adaptability: This solutions is highly adaptable to new environment. All the user needs to do is provide new occupancy grid map of the floor plan, and beacon reception maps of the beacons deployed in the floor.

6 CONCLUSION

In conclusion, a particle filter based Indoor localization solution was designed and developed. The solution shows good accuracy by making use of simple off-the-shelf sensors and resources. Hence making this solution highly scalable to various applications. Some use cases and applications of this system can be in supermarkets or shopping malls, where the heat map functionality can be used to identify busy locations that can be used to derive information about the mall/supermarket. based on these hotspots, a recommendation system could also suggest to owners where to place advertisement or stalls for maximum exposure. This can also recommend to customers the popular areas in the mall, or centers of attraction. This solution can also be implemented in hospitals, where patients movement and daily activities can be monitored. Several other applications like Musuem recommendation or in paint ball gaming, where it could be used for surveillance or user's radar. Some Future works that can be done to improve this solution are:

- Improve beacons: the beacons used in this solution would have a high influence in the estimation. Better beacons with good signal strengths can allow for the use of RSS values to weight the particles or also use triangulation as alternative estimation technique.
- Improve Sensors: improvement in the heading estimation would improve the estimation in general. Methods that use accelerometer data for compensation of tilt movements would help in reducing the errors induced in tilts. Step detection is good, but to increase the scope and reliability, gait data and analysis of more people should be collected
- Structure: There is a poor distribution of workload in the system in general. It would be beneficial to move bulk of the processing to the cloud and make use of wifi-enabled boards. Where this would make the process much more easier and convenient.

REFERENCES

[1] Adafruit. Adafruit adafruit lsm9ds1 accelerometer + gyro + magnetometer 9-dof breakout.

[2] Blueup. Blueup bluetooth low energy beacons and sensors for the internet of things.

[3] GUSTAV EHRENBORG and HERMAN FRANSSON. Sensor fusion based indoor positioning with ibeacons.

[4] Zahid Farid, Rosdiadee Nordin, and Mahamod Ismail. Recent advances in wireless indoor localization techniques and system. *Journal of Computer Networks and Communications*, 2013, 2013.

[5] Nammoon Kim and Youngok Kim. Indoor positioning system with imu, map matching and particle filter. *Recent Advances in Electrical Engineering and Control Applications; Springer: Berlin, Germany*, pages 41–47, 2015.

[6] Dalila Ksentini, A Rahim Elhadi, and Noureddine Lasla. Inertial measurement unit: evaluation for indoor positioning. In *Advanced Networking Distributed Systems and Applications (INDS), 2014 International Conference on*, pages 25–30. IEEE, 2014.

[7] Binghao Li, Andrew Dempster, Chris Rizos, and Joel Barnes. Hybrid method for localization using wlan. In *Spatial Sciences Conference*, pages 341–350. Citeseer, 2005.

[8] Mbed. Mbed nordic nrf51-dk.

[9] Chris Rizos, Andrew G Dempster, Binghao Li, and James Salter. Indoor positioning techniques based on wireless lan. 2007.

[10] Arno Solin, Santiago Cortes, Esa Rahtu, and Juho Kannala. Inertial odometry on handheld smartphones. *arXiv preprint arXiv:1703.00154*, 2017.

[11] Zeynep Turgut, Gulsum Zeynep Gurkas Aydin, and Ahmet Sertbas. Indoor localization techniques for smart building environment. *Procedia Computer Science*, 83:1176–1181, 2016.