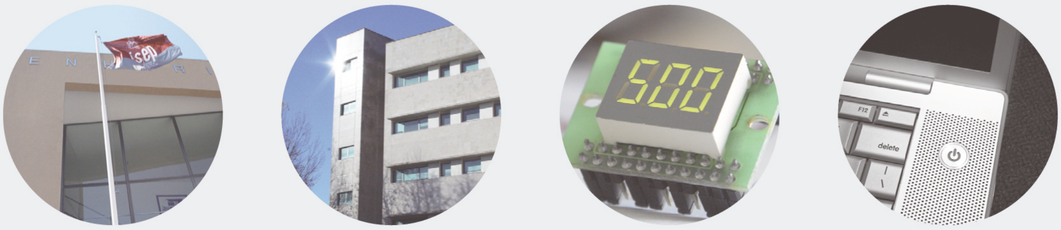
INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO

MESTRADO EM ENGENHARIA ELECTROTÉCNICA - SISTEMAS ELÉCTRICOS DE ENERGIA



INDOOR POSITIONING SYSTEM SURVEY USING BLE BEACONS

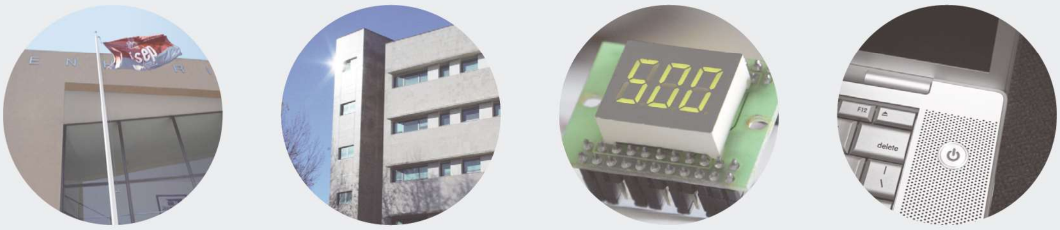
MATHAN KUMAR BALAKRISHNAN

novembro de 2017

POLITÉCNICO
DO PORTO

INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO

MESTRADO EM ENGENHARIA ELECTROTÉCNICA - SISTEMAS ELÉCTRICOS DE ENERGIA



INDOOR POSITIONING SYSTEM SURVEY USING BLE BEACONS

MATHAN KUMAR BALAKRISHNAN

Outubro de 2017

POLITÉCNICO DO PORTO

INDOOR POSITIONING SYSTEM SURVEY USING BLE BEACONS

MATHAN KUMAR BALAKRISHNAN

Instituto Superior de Engenharia do Porto

Department of Electrical Engineering

Masters in Electrical Engineering – Power Systems

2017

Report elaborated for partial fulfillment of the requirements of the DSEE curricular unit – Dissertation for the Master degree in Electrical Engineering – Electrical Systems of Energy

Candidate: *Mathan Kumar Balakrishnan*, Nº 1150253, email 1150253@isep.ipp.pt

Scientific advisor: *Jorge Manuel Estrela da Silva*, jes@isep.ipp.pt

Company: Creative Systems, C.A, São João da Madeira, Portugal

Company supervisor: *João Sousa*, email jsousa@tycoint.com

Instituto Superior de Engenharia do Porto

Department of Electrical Engineering

Masters in Electrical Engineering – Power Systems

2017

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to my advisor **Prof. Jorge Estrela da Silva**, **Instituto Superior de Engenharia Do Porto** for the continuous support of my Master thesis, for his patience, motivation, enthusiasm, and immense knowledge. I could not have imagined having a better advisor and mentor for my project. The door to **Prof. Jorge** office was always open whenever I ran into a trouble spot or had a question about my research or writing. He consistently allowed this paper to be my own work, but steered me in the right direction whenever he thought I needed it.

I would also like to acknowledge **Prof. Teresa Alexandra Nogueira Ph.D., Course Director**. Besides my advisor, at **Instituto Superior de Engenharia Do Porto**, I would like to thank the rest of my thesis committee: **Mr. João Sousa**, Head of Software Development departments and Thesis Supervisor, Creative Systems, São João Madeira and **Mr. João Oliveira** Team Leader and Software Developer, Creative Systems, for their encouragement, insightful comments, and hard questions. With great pleasure my profound sense of gratitude to R&D, Other department Engineers, and Co-workers of Creative Systems, São João Madeira, for their valuable support and encouragement to do this Master's Thesis.

Last but not the least, I would like to thank my family, for giving birth to me at the first place and supporting me spiritually throughout my life. I must express my very profound gratitude to my friends, for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

ABSTRACT

This project provides a survey of indoor positioning systems and reports experimental work with Bluetooth Low Energy (BLE) Beacons.

A positioning algorithm based on the Received Signal Strength Index (RSSI) from Bluetooth Low Energy signals is proposed for indoor tracking of the position of a drone. Experimental tests for characterization of beacon signals are presented. The application of a Kalman filter to reduce the effect of fluctuations in beacons signals is described.

Keywords

Bluetooth Low Energy, Indoor Positioning Systems, RSSI

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

AP	-	*ACCESS POINT*
BER	-	*BIT ERROR RATE*
BLE	-	*BLUETOOTH LOW ENERGY*
GPS	-	*GLOBAL POSITIONING SYSTEM*
HW	-	*HARDWARE*
IPS	-	*INDOOR POSITIONING SYSTEM*
LED	-	*LIGHT EMITTING DIODE*
LNMS	-	*LOG NORMAL SHADOWING MODEL*
LQ	-	*LINK QUALITY*
PDU	-	*PROTOCOL DATA UNIT*
RFID	-	*RADIO FREQUENCY IDENTIFICATION*
RSSI	-	*RECEIVED SIGNAL STRENGTH INDEX*
SDK	-	*SOFTWARE DEVELOPMENT KIT*
SW	-	*SOFTWARE*
TPL	-	*TRANSMITTED POWER LEVEL*
TX	-	*TRANSMISSION POWER*
UI	-	*USER INTERFACE*
USB	-	*UNIVERSAL SERIAL BUS*
UUID	-	*UNIVERSALLY UNIQUE IDENTIFIER*

WLAN - *WIRELESS LOCAL AREA NETWORK*

1. INTRODUCTION

1.1 MOTIVATION

The desire and need for indoor positioning systems (IPS) is becoming more widespread in the market. Wireless indoor positioning systems have become very popular in recent years. These systems have been successfully used in many applications such as asset tracking, airports, museums, inventory management and so on.

Be it a desire to find the position of a drone or any object, the need to locate a seat inside a large arena or simply the wish to learn more about objects in ones surroundings, there is an increasing demand for indoor positioning systems. Global positioning system (GPS) has proved to be highly reliable and accurate when used in the context of outdoor localization. However, when used for indoor localization, the detection and decoding of GPS signals is a difficult task due to the additional signal attenuation caused by physical obstacles such as the walls of buildings. Due to the limitation of GPS in an indoor setting, there have been many technologies proposed and discussed in literature that pose as alternatives for indoor localization.

These technologies and work pertaining to them are discussed below. Despite much research into alternatives to GPS, none of these technologies have made a substantial impact in the field such that one could be claimed as a standard for indoor localization as GPS can be claimed for outdoor. Bluetooth Low Energy has been recognized as one such technology with high potential to perform indoor localization and as such will be the focus of this project [10].

The outcomes of this project are intended to assist the feasibility analysis of the use of current Bluetooth LE technology in the context of indoor localization to find a position of a drone or any object.

1.2 CONTEXTUALIZATION

This thesis was motivated by recent developments at Creativesystems. Creativesystems is a company that develops integrated solutions for data flow automation and optimization, supported by expert consulting in innovation, operational management and interactive experiences.

Figure 1- Creativesystems, São João da Madeira, Portugal

Creativesystems creates and implements solutions for identification and automatic traceability that cover all life cycle of projects (design, hardware, software, services, support), with special emphasis in sectors of retail, logistics and industry.

The company is present at Germany, Brazil, Dubai and Portugal, and it has projects in the USA, United Kingdom, Denmark, Spain, Russia, Belgium, Colombia and South Africa.

Creativesystems recently started a project called “RFID Inventory using Autopilot Drone” The project is divided into five components, being one of those “INDOOR POSITIONING SYSTEM USING BLE BEACONS”.

The components are,

1. Indoor Positioning System
   a. To find the position of the Drone in indoor.
2. RFID Reading Module
   a. Hardware component to manage RFID reader and antennas
   b. Software component to handle captured reading and send them to Middleware
3. Mobile App for Drone Flight Control and Flight Control Module
4. Middleware
   a. Logic: Get, Save and Send inventory data to TrueVUE Inventory Intelligence Software
   b. UI: Application to manage drones, flight routes, and check inventory information
5. Mobile App for Inventory Dashboard

From these modules, this thesis deals with the Indoor Positioning System. The objective of the company in this component is to develop a positioning system to find the position of a drone indoors.

One of the envisioned purpose for the drone is to read the RFID tags in warehouse, shops and so on. Then send all the information's or data to the server. It can be used in the warehouse or in the shopping complex where there is a need for this purpose and also to reduce human intervention.

The information that drone sends to the server contains, (for example, product expiry date, product price, number of products sold or, number of products remaining) The drone contains RFID reader, antenna, Raspberry Pi board, fixed in it. The RFID reader obtains the tags ID and saves it in the memory connected externally. Once the readings are done by the drone, it sends the information to the server.

1.3 OBJECTIVES

The main objective of this thesis is to provide a survey of currently available technology for indoor positioning with emphasis on Bluetooth technology. This study includes the choice of a possible solution from commercially available systems and experimental tests to assess the feasibility of the proposed solution. The solution should be based on low cost technology and should be suitable for integration in small drone for indoors operation.

1.4 THESIS STRUCTURE

Chapter gives the explanation regarding company, motivation to choose this project, also about the objective of the thesis. Chapter 2 deals with the trilateration method and Kalman filtering algorithm, where the equations are described in detail. Chapter 3 presents an overview of wireless communication protocols, namely Wi-Fi, ZigBee, Bluetooth and Bluetooth Low Energy. On chapter 4, commercially available beacons like Estimote, Kontakt, Paypal, Ibeacon are discussed and comparison made for choosing the beacons for this project. Chapter 5 deals with the computational system, structure of the proposed model, and the experimental setup for the tests. The experimental test results and error analysis are presented on chapter 6. Finally, the thesis is concluded in this chapter by using previous results and the future work ideas were discussed.

The timeline of the thesis work is given below with brief description:

- **PLANNING** – Initially the planning of the project was scheduled in a team meeting and also about the description of the project.

- **RESEARCH AND ANALYZING FRAMEWORK** – Choosing the hardware to be implemented in the project, and to find the best hardware which is commercially available in the market.

- **INVETIGATE ALGORITHM FOR FINDING A POSITION** – Research about the filtering algorithm to be implement and to test the algorithm by testing experiments.

- **IMPLEMENTATION OF HARDWARE AND SOFTWARE** – Configuring beacons and receive the RSSI signals. Also, to synchronize the hardware and software.

- **EXPERIMENTATION** – Practical analysis using the experimental tests by choosing a room and mesh points with the deployment of beacons was already done.

- **TESTING AND PROBLEM SOLVING** – Finally, analyzing the results and if there is any problems, the future work is to be discussed.

Table 1 presents the timeline of this project in detail.

	Task	tart	End	Days
1	**Planning**			
1.1	Team Meeting	Wed 01/03/17	Wed 01/03/17	1
1.2	Analyze about Hardware	Thu 02/03/17	Thu 02/03/17	1
1.3	Meeting with the team	Thu 02/03/17	Thu 02/03/17	1
1.4	Scheduling Plan	Fri 03/03/17	Fri 03/03/17	1
1.5	Description of the project	Mon 06/03/17	Fri 10/03/17	5
2	**Research and Analysing Framework**			
2.1	Architecture Diagram and Analyzing Beacons	Mon 13/03/17	Tue 21/03/17	7
2.2	Investigate Beacons in the Market	Wed 22/03/17	Mon 27/03/17	4
2.3	Investigate Implementation process	Tue 28/03/17	Fri 07/04/17	9
2.4	Analyzing about Algorithm and filter	Mon 10/04/17	Wed 19/04/17	8
3	**Investigate Algorithm for finding a position**			
3.1	Kalman filter and trilateration algorithm	Thu 20/04/17	Fri 28/04/17	7
3.2	Testing the Algorithm	Mon 01/05/17	Fri 05/05/17	5
4	**Implementation HW and SW**			
4.1	Configure Beacons	Mon 08/05/17	Tue 09/05/17	2
4.2	Integrating Raspberry pi and Beacons	Wed 10/05/17	Tue 16/05/17	5
4.3	Synchronizing Hardware and software	Wed 17/05/17	Fri 19/05/17	3
5	**Experimentation**			
5.1	Practical Analysis using HW and SW	Mon 22/05/17	Wed 24/05/17	3
6	**Testing and Problem Solving**			
6.1	Implementation and Result	Thu 25/05/17	Wed 31/05/17	5

Table 1 - Project chronogram

2. POSITIONING FUNDAMENTALS

2.1 INTRODUCTION

This chapter discusses two frequently techniques used in positioning: the trilateration method and filtering. Although several other filtering and data fusion techniques exist, the Kalman filter, being the most classic approach, is chosen as the object of study. Finally, the basic concepts for estimating distance from radio frequency signal attenuation are presented.

2.2 TRILATERATION

Trilateration is the process of determining absolute or relative locations of points by measurement of distances, using geometry of circles, spheres or triangles. Trilateration is used, for instance, in the Global Positioning System (GPS).

Trilateration uses the distances to a set of known fixed points to estimate the current position. It is possible to use trilateration for determining a position in both two and three dimensions. In two dimensions it is a problem of finding the intersection between a certain number of circles. In three dimensions, it changes into a problem of finding the intersection between a certain number of spheres.

In the two-dimensional case, given the exact distance to three reference points and their position, it is possible to determine the current position (Figure 5).

In the three-dimensional case, each reference node forms a sphere around itself with the radius corresponding to the distance from it to the current position. In the general case, the position can be determined from the intersection of four spheres

It is not always possible to measure the distance between two objects explicitly. For example, in the case of GPS, the receiver measures the time it takes for the signals to move from the satellites to the GPS receiver. The distances can then be calculated since the velocity of the radio signals is known.

Moreover, the measurements are subject to noise. In the indoors case, some technologies, such as Ultra-wide band, Bluetooth, Wi-Fi are more suitable for this method than others [25].

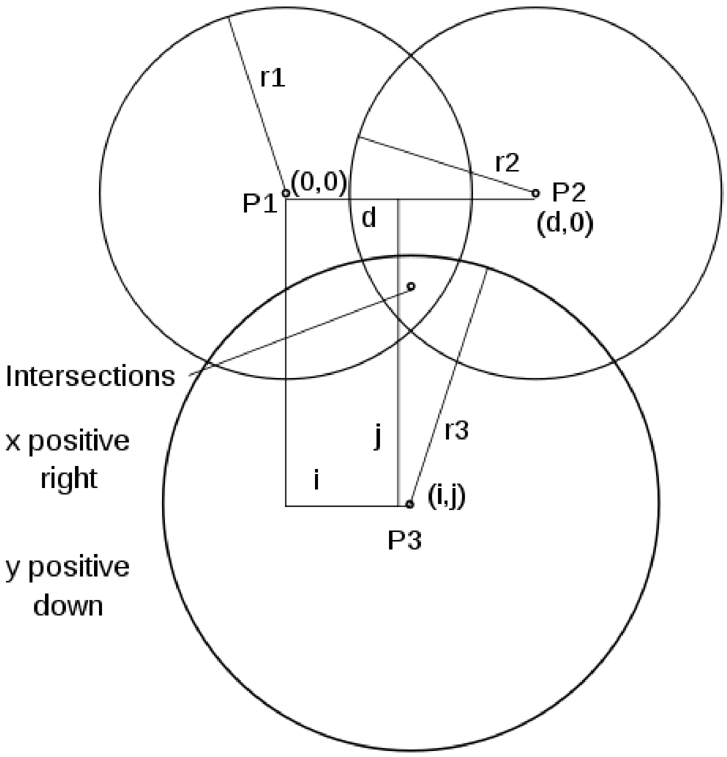


Figure 2- Trilateration

The position of the Bluetooth device is found by formulating the equations for the three spheres centered at each beacon and radius corresponding to the estimated distance, and then solving the three equations for the three unknowns, x, y, and z. To simplify the calculations, the equations are formulated assuming the beacons are on the $z = 0$ plane. Also, the formulation is such that one center is at the origin, and one other is on the *x*-axis. It is possible to formulate the equations in this manner since any three non-collinear points lie on a unique plane.

The equations for the three spheres are as follows:

$$\mathbf{r_1^2 = x^2 + y^2 + z^2} \quad \mathbf{(1)}$$

$$\mathbf{r_2^2 = (x-d)^2 + y^2 + z^2} \quad \mathbf{(2)}$$

$$\mathbf{r_3^2 = (x-i)^2 + (y-j)^2 + z^2} \quad \mathbf{(3)}$$

The value *d* is the x coordinate of point P2. It needs to be subtracted from x to get the length of the base of the triangle between the intersection and r2 (*x*, *y*, *z* are coordinates, not lengths).

Subtracting both sides of equations 1 and 2, we get

$$\mathbf{r_1^2 - r_2^2 = x^2 - (x-d)^2} \quad \mathbf{(4)}$$

$$\mathbf{r_1^2 - r_2^2 = x^2 - (x^2 - 2xd + d^2)} \quad \mathbf{(5)}$$

$$\mathbf{r_1^2 - r_2^2 = 2xd - d^2} \quad \mathbf{(6)}$$

$$\mathbf{r_1^2 - r_2^2 + d^2 = 2xd} \quad \mathbf{(7)}$$

So, x is given by

$$\mathbf{x = (r_1^2 - r_2^2 + d^2)/ 2d} \quad \mathbf{(8)}$$

Substituting the expression for *x* back into the equation for the first sphere produces the equation for a circle, the solution to the intersection of the first two spheres:

$$\mathbf{y^2 + z^2 = (r_1^2) - (r_1^2 - r_2^2 + d^2)^2 / 4d^2} \quad \mathbf{(9)}$$

Substituting $\mathbf{z^2 = r_1^2 - x^2 - y^2}$ into the formula for the third sphere and solving for *y* there results:

$$\mathbf{y = (r_1^2 - r_3^2 - x^2 + (x-i)^2 + j^2)/ 2j} \quad \mathbf{(10)}$$

$$\mathbf{y = (r_1^2 - r_3^2 + i^2 + j^2) / 2j) - (i/j * x)} \quad \mathbf{(11)}$$

Now that the *x*- and *y*-coordinates of the solution point are found, the formula can be rearranged for the first sphere to find the *z*-coordinate:

$$\mathbf{z = \pm \sqrt{r_1^2 - x^2 - y^2}} \quad \mathbf{(12)}$$

Now the solution to all three points *x*, *y* and *z* is found. Because *z* is expressed as the positive or negative square root, it is possible for there to be zero, one or two solutions to the problem.

This last part can be visualized as taking the circle found from intersecting the first and second spheres and intersecting that solution with the third sphere. If this falls entirely outside or inside the first solution, then *z* is equal to the square root of a negative number, meaning no real solution exists. If it touches the first solution at exactly one point, *z* is equal to zero and if it touches at two points, then *z* is equal to plus or minus the square root of a positive number.

The derivation above assumes that the coordinate system in which the sphere centers are designated must be such that

1. all three centers are in the plane $z = 0$,
2. the sphere center, *P*1, is at the origin, and
3. the sphere center, *P*2, is on the *x*-axis.

In some situations, it might be convenient to define the coordinate system differently.

This problem can be overcome as described below where the points, *P*1, *P*2, and *P*3 are treated as vectors from the origin where indicated. *P*1, *P*2, and *P*3 are expressed in the original coordinate system.

$\mathbf{e_x = (P2 - P1) / \|P2 - P1\|}$ is the unit vector in the direction from *P*1 to *P*2 where||.|| is the Euclidean norm:

$$\mathbf{\|P2 - P1\| = sqrt((P2_x - P1_x)^2 + (P2_y - P1_y)^2)}$$

- i = $\mathbf{e_x(P3 - P1)}$ is the projection of vector from *P*1 to *P*3 onto EX.

$\mathbf{e_y = (P3 - P1 - i \cdot e_x) / \|P3 - P1 - i \cdot e_x\|}$ is the unit vector in the y direction. Note that the points *P*1, *P*2, and *P*3 are all in the $z = 0$ plane of the figure coordinate system.

$\mathbf{d = \|P2 - P1\|}$ the distance between the centres *P*1 and *P*2 and

$\mathbf{j = e_y(P3 - P1)}$ is the signed magnitude of the y component, in the figure coordinate system, of the vector from $P1$ to $P3$.

Using **i**, **d**, and **j** as computed above, solve for x, y, and z as described in the derivation section

In order to represent the location of the drone in the original coordinate system, the (x,y) coordinates must be transformed back to the original coordinate system.

In order to achieve this, define X_w as the coordinates of a vector in a given Cartesian coordinate system, and X_b as the coordinates of the same vector in the Cartesian coordinate system with origin in P1 and the x axis oriented with P2-P1.

The transformation from the former to the latter coordinate system is computed as follows:

$$\mathbf{X_b = T(X_w - P1)}$$

where

$$\mathbf{T} = \begin{bmatrix} \frac{\boldsymbol{P2-P1}}{\|\boldsymbol{P2-P1}\|} \\ \frac{\boldsymbol{P3-P1-i.e_x}}{\|\boldsymbol{P3-P1-i.e_x}\|} \end{bmatrix}$$

is a transformation matrix defined using the arguments above. Since the T matrix is orthogonal, the inverse of T is its transpose T^T. Therefore,

$$\mathbf{X_w = P1} + \begin{bmatrix} \frac{\boldsymbol{P2-P1}}{\|\boldsymbol{P2-P1}\|} \\ \frac{\boldsymbol{P3-P1-i.e_x}}{\|\boldsymbol{P3-P1-i.e_x}\|} \end{bmatrix}^{\boldsymbol{T}} \mathbf{X_b}$$

2.3 KALMAN FILTER

Multi-path reflection, meaning that the signals bounce against objects in the environment, is a major factor influencing distance estimation. To reduce the effect of the noise on the estimated distances a Kalman filter can be used. The Kalman filter uses the history of measurements to make estimations of the next RSSI [2].

A transition model is used to model the evolution from time step t-1 to time step t:

$$\boldsymbol{x_{t+1} = A_t x_t + B_t \mathbf{U}_t + \epsilon_t} \quad \mathbf{(13)}$$

where x_t is the current state, x_{t+1} is the state at the next time step, A_t is a transformation matrix, U_t is a control input and ϵ_t is the process noise, assumed to follow a normal distribution with covariance R_t. The process noise is used to model both the effect of external disturbances and model imperfections.

In what follows, a system with no control input is considered and A set to the identity matrix. Changes in the system state will be modelled as process noise. These two changes result in the following simplified model:

$$\boldsymbol{x_t = x_{t-1} + \epsilon_t} \quad \textbf{(14)}$$

To define how a state x results in a measurement (or observation) z, the following model is considered:

$$\boldsymbol{z_t = C_t x_t + \delta_t} \quad \textbf{(15)}$$

where C_t is the transformation matrix and δ_t is measurement noise, assumed to follow a normal distribution with covariance Q_t. In what follows, it is assumed that the measurements correspond directly to the state. This results in the following simplified measurement model:

$$\boldsymbol{z_t = x_t + \delta_t} \quad \textbf{(16)}$$

The Kalman filter involves two operations, prediction and update. In the prediction step, performed at each time step, the most likely next state is calculated according to the knowledge about previous measurements and noise levels. When the prediction μ_t $(x_{t|t-1})$ is made, there is also an estimation of how likely the prediction is to be correct. This is represented by the uncertainty Σ (error covariance matrix). Under the considered assumptions, the prediction step is described by:

$$\boldsymbol{\bar{\mu}_t = \mu_{t-1}} \quad \textbf{(17)}$$

$$\boldsymbol{\bar{\Sigma}_t = \Sigma_{t-1} + R_t} \quad \textbf{(18)}$$

In the update step, the estimated state is updated with measurement information. This update is based on the value the Kalman gain K:

$$\boldsymbol{K_t = \bar{\Sigma}_t\, (\bar{\Sigma}_t\, Q_t)^{-1}} \quad \textbf{(19)}$$

The Kalman gain weighs between the certainty of the prediction and the certainty of the measurement. For example, if the measurement has a high certainty, whereas the prediction has low certainty (in the case of high process noise or absence of measurements), the measurement will have more impact in the estimated state than the prediction. Finally, the estimated state and are given by:

$$\mu_t = \bar{\mu}_t + K_t (z_t - \bar{\mu}_t) \quad (20)$$

$$\Sigma_t = \bar{\Sigma}_t - (K_t \bar{\Sigma}_t) \quad (21)$$

Therefore, if the system state is known to change slowly, then R can be set accordingly to a low value.

2.4 ESTIMATING DISTANCE FROM SIGNAL ATTENUATION

Ideally, the distance between the receptor of a radio frequency signal and the source of that signal can be computed from the observed attenuation of the signal. This implies that both the transmitted power and the signal power at the receptor are known or measured.

In practice, the transmitted signal is subject to several phenomena that attenuate the signal power. Path loss is the "attenuation of signal power of a signal as it propagates through space". Likely causes of path loss include:

- **Free Space Loss:** the degradation of a signal due to distance (assuming complete line of sight between transmitters and receivers)
- **Fading Loss:** time variation of signals due to changes in the transmission channel as a result of things such as furniture, human interference, movement of the devices.
- **Multipath losses** (reflection, diffraction, scattering): the signal is scattered or reflected due to objects changing its phase, attenuation or delay resulting in a degraded signal. Causes of multipath losses include reflection, diffraction and scattering.
- **Refraction:** change in the direction of an electromagnetic way as a result of changes in the medium in which the signal travels.
- **Noise and interference:** unwanted signals within the medium that affect the signals by distorting them at the receiver [20].

In general, radio frequency signal strength measures are expressed in dBm. The following formula establishes the relationship between the received signal strength (RSSI, in dBm) and distance (in meters) [3, 8]:

$$d = 10^{\wedge}((P - RSSI)/(10 \cdot n))$$

where:

d = Distance between the transceiver and recipient,

P = Signal Strength measured at a distance of 1 meter of the device,

RSSI = Signal from the Beacons,

n = Signal propagation exponent.

Environment	**Path Loss Exponent** n
Free Space	2
Urban Area Cellular Radio	2.7–3.5
Shadowed Urban Cellular Radio	3–5
Line-of-Sight in Building	1.6–1.8
Obstruction in Building	4–6
Obstruction in Factories	2–3

Table 2 - Path loss exponents for different environments

3. WIRELESS COMMUNICATION STANDARDS

3.1 INTRODUCTION

Many technologies have been proposed for use in indoor positioning systems. However, there has not been enough proof or evidence to declare a single technology as a standard for indoor localization, such as GPS has become the standard for outdoor localization until now. However, GPS is impractical in the context of indoor localization. Consequently, alternative communication technologies have been proposed and discussed in this survey [11].

Some of the proposed alternatives for indoor positioning systems are: Wi-Fi, ZigBee, Ultra-wideband radio, Traditional Bluetooth and Bluetooth Low Energy. However, the technologies that have caught the most attention in the field of indoor positioning are Bluetooth and Wi-Fi. Thus, background information and testing analysis are presented and discussed the following sections. For the purposes of this project, Bluetooth LE has been determined as the most suitable communication protocol for use in the developed IPS mainly because of its compatibility, less consumption of battery power, easily deployment and high range availability [6, 10].

3.2 ZIGBEE

ZigBee is a wireless communication standard developed by the ZigBee Alliance. It was proposed to specifically address the need for low-cost implementation of low-data-rate wireless networks with ultralow power consumption.

Due to its energy saving and improved security, ZigBee technology was originally intended for applications like home automation (remote lights and thermostat monitoring and control), urban traffic light control, health care, and agriculture, among many others In addition, ZigBee has been used to develop indoor positioning systems because it is a low-cost, low-power consumption technology and because it is easy to obtain RSSI levels as these are incorporated in each of the packets sent, with no additional hardware needed.

An indoor positioning system based on ZigBee is composed of a network of sensors and wireless sensor network algorithms. Most of the algorithms used in these systems use the RSSI values to estimate the location, relying thus on the same techniques as Wi-Fi and Bluetooth, that is, fingerprinting and propagation models. A commercial project using ZigBee is Netvox (website: http://www.netvox.com.tw), in which location is part of a complete home automation platform [30].

3.3 Wi-Fi

Wi-Fi communication is based on the transmission of radio waves at frequencies of 2.4 GHz or 5 GHz. A study on the effects of indoor localization techniques in a Wi-Fi access point (AP) intense environment was presented in. In particular, the algorithms explored throughout this paper may prove useful when applied in a Bluetooth LE context. This paper alludes to some factors that may need to be considered throughout this project that have the potential to affect the accuracy of localization. These include: the effect of a large amount of APs (or transmitters), time tolerance i.e. signal strength decreasing over time, APs disappearing and appearing, different mobile device specifications and orientation of devices, etc. [10]

3.4 BLUETOOTH

Bluetooth is a technology that has received much attention and usage in the field of indoor positioning. It is designed to enable short range wireless communication between devices using radio signals in the 2.4 GHz range. It has been designed to support low power wireless communication and hence, power control is one of the better and desirable features associated with it. One major advantage attributed to the use of Bluetooth technology is high penetration in consumer products and society in general. In order to achieve this, the required hardware is mass produced thus making Bluetooth technology very accessible.

In 2010, Bluetooth Low Energy (Bluetooth LE), described in the next section, was introduced along with the specification for Bluetooth 4.0 technology. It has grown up massively during 2016 and many companies are manufacturing in a mass production [8],[21],[22].

The power control feature allows a transmitter to adjust its strength based on the Received Signal Strength Indicator (RSSI) from another device. RSSI is one of the most frequently used measures for distance estimation [4],[5],[19],[25]. Other available measures are Link Quality (LQ) which can be picked up by a Bluetooth receiver and the transmitted power level (TPL) which is transmitted. The next sub-sections describe these signal strength parameters in more detail.

3.4.1 RECEIVED SIGNAL STRENGTH INDICATOR

The Received Signal Strength Indicator (RSSI) parameter is a signed 8-bit integer value that provides an indication of the signal strength experienced by the receiver of a Bluetooth Protocol Data Unit (PDU). For Bluetooth Low Energy capable devices, the RSSI range is typically between -127 to 20 dBm, where a larger value indicates a stronger signal and lower will be weaker signals.

In [27], a novel approach to determine the location of a mobile node using mobile and fixed wireless beacons is proposed. The proposed approach has the following contributions:

- Established a new system to model the localization with RSSI
- Smartphone based system which is cost effective and easy to use.
- System is able to protect user privacy
- Comparison of system accuracy with mobile and fixed wireless node (Beacons).

System	Environment	Accuracy	Percentage
Android	Indoor	< 2.0 meters	85%
	Outdoor	< 1.5 meters	90%
iPhone	Indoor	< 2.5 meters	80%
	Outdoor	< 1.8 meters	90%

Table 3 Accuracy of the developed system

In [28], RSSI based distance measurement model was tested using TelosB motes with CC2420 radio for distance ranging from 1m to 8m. The mean distance error for indoor environment obtained is 2.249m which is due to absence of calibration. Similar to RSSI based distance estimation has been done for IRIS motes with Atmel's AT86RF230 radio for a 20m x 20m indoor and outdoor environment. In order to minimize localization error, Log Normal Shadowing Model (LNMS) and calibration equation has been employed [19].

Based on the experimental results it is claimed that the mean distance error in the indoor environment is 0.9753m. In [29], the authors have shown that there exists RSSI nonlinearity for well-known 802.15.4 radios namely CC2420 and AT86RF230 radios. Furthermore, in the paper non-linear RSSI response curves for the two radios in indoor environment have been derived experimentally and a calibration method is used to get rid of non-linearity in the two radios. It is found that RSSI values differ for different radios for a known distance and the selection of RSSI based localization depends on the accuracy requirement of the application.

In RSSI based localization technique, performance degradation is mainly due to interference caused by multiple devices such as Wi-Fi routers and Zigbee nodes operating in the same frequency [14]. In order to study the impact of interference on indoor localization, a benchmark has been proposed in [14].

3.4.2 LINK QUALITY (LQ)

The Link Quality (LQ) parameter is an unsigned 8-bit integer value which is most commonly related to the average bit error rate (BER). Hence, the LQ parameter spans the integer range 0 to 255, where a higher number indicates a better link.

3.4.3 TRANSMITTED POWER LEVEL (TPL)

The Transmitted Power Level (TPL) is represented by a signed 8-bit integer. TPL is measured in dBm and as such will have a maximum value of +20dBm. The minimum value is device specific. The idea behind TPL is that Bluetooth chips with Transmitter (Tx) power between +4 dB and +30 dB are required to perform power control, where Bluetooth chips with high output power scale down the Tx effect to save power when the link is good.

3.5 BLUETOOTH LOW ENERGY

Bluetooth Low Energy, as the name implies, was primarily designed to feature lower power consumption than general Bluetooth technology while maintaining a similar communication range. This attribute was ideal and important for this project as amongst the other trade-offs, power consumption and battery life of a Bluetooth LE enabled device have a direct correlation [1].

It is claimed that the low power consumption of Bluetooth LE capable devices means they can potentially last for years powered by a single coin cell battery. This is an important advantage and condition to meet the requirements of the project as small and easily deployed transmitters that require minimal setup are needed within the system.

Technologies based on Bluetooth LE technology have been used to provide a means for Bluetooth LE enabled devices to communicate through signals contains packets of data. These are small packets of data that are broadcasted at regular intervals by transmitters (beacons). Moreover, traditional Bluetooth is more appropriate for systems that require large data transfers whereas Bluetooth LE is more suited for applications that require small amounts of data [12].

There are some aspects that need to be considered when choosing technologies based on Bluetooth LE. The first aspect is the time interval between transmissions. Bluetooth LE devices generally transmit 10 signals per second maximum and are able to change the time interval in the beacon settings. This interval has a direct correlation to the time it takes to discover a device. Additionally, the transmit range of Bluetooth LE based transmitters has a direct correlation with the Transmission power level (TPL). At the highest transmission rate, the largest possible signal coverage can be obtained. However, this comes at the cost of battery life. This is another important point that must be taken into account when using Bluetooth LE based transmitters for indoor localization [8].

In general, Bluetooth LE enabled devices do not support reception of the LQ parameter, hence, removing the possibility of using this parameter within this project. Transmitter technologies based on Bluetooth LE typically have a fixed transmission power level (TPL) implying that the use of the TPL parameter is also no longer a viable option. Fortunately, the RSSI parameter is still easily accessible in Bluetooth LE enabled devices via simply receiving a broadcasted message.

It is claimed to be the parameter within Bluetooth that has received the most attention and consensus to be the best suited for positioning applications despite its flaw in not being optimal. Based on this, the RSSI parameter (in which Bluetooth LE capable devices can receive and can be used in positioning algorithms to infer location) will be the primary focus during the development of the IPS in this project [23].

3.5 SUMMARY

Table 3 presents a summary of the main characteristics of the above mentioned wireless communication standards. The main contenders are Wi-Fi and Bluetooth technologies. Although Wi-Fi can be used in a similar way as BLE beacons, with extra coverage due to stronger signal, in general it requires an external power source, more setup costs and pricier equipment. [9].

	ZigBee	Wi-Fi	Bluetooth	BLE
Network topology	Star, cluster, or mesh	*Ad-hoc*, or Star	Scatternet	Star-bus
Frequency Band	868 MHz (Europe) 915 MHz (North America) 2.4 GHz	2.4/5 GHz	2.4 GHz	2.4 GHz
Data Rate	250 Kbps	11/54 Mbps	1 to 3 Mbps	1 Mbps
Range	10 to 100 m	Up to 100 m	Up to 10 m	Up to 40 m
Power Consumption	Very low	High	Low	Very low
Battery Life	Multiple years	Multiple hours	Multiple weeks	Multiple months
Cost	Low	High	Medium	Low
Infrastructure	To be deployed	Existing Wi-Fi nodes	To be deployed	To be deployed
Smartphones	Not supported	Supported	Supported	Supported
Developed for positioning	No	No	No	Yes
Accuracy	3-5 m	5-10 m	2-5 m	1-2 m
Typical applications	Industrial control and monitoring, sensor networks.	WLAN, broadband connections.	Inter-device data transfer (i.e. cable replacement).	Sensors, positioning, peripherals.

Table 4 Comparison of ZigBee, Bluetooth, BLE and Wi-Fi technologies

4. BLUETOOTH BEACONS

4.1 INTRODUCTION

The following requirements must be satisfied when selecting the technology to be used for the transmitters and receivers of this IPS:

1. Based on Bluetooth Low Energy technology,
2. Small and easily deployed,
3. Require minimal calibration and setup at the deployment location,
4. Able to provide location based information,
5. Developed for iOS (and Android, where possible).

The commercial technologies: iBeacon, Estimote, Kontakt, and PayPal Beacon have been explored below.

4.2 ESTIMOTE

Estimote Beacons broadcast depending on the implementation, devices could probe the signal every second (1 Hz) or 10 times a second (10 Hz), tiny radio signals based on Bluetooth LE (requirement 1 above). Estimote is based on iBeacon technology and the Estimote SDK allows apps on smart phones (iOS) to understand their proximity to nearby objects and locations, recognizing location based information (requirements 4 and 5).

However, unlike iBeacon, Estimote is not completely commercialized as they are still in a development phase. Due to this, this technology was looked upon less favorably in the final decision process. Another disadvantage is that it can only be used in iOS and not in Android yet.

4.3 PAYPAL BEACON

PayPal Beacon uses Bluetooth LE as its underlying communication technology (requirement 1). It requires a connection to a Smartphone in order to be authenticated by a remote PayPal server. However, PayPal fails to meet the requirement of requiring minimal calibration and setup at the deployment location (requirement 3) as the transmitter also requires Wi-Fi technology. Furthermore, PayPal Beacons are more focused on making purchases (in sales) without the need for cards, cash or signatures instead of for determining location. Consequently, this technology was deemed as not being suitable for this project.

Based on these findings iBeacon technology was the Commercial product that stood out the most and met all the requirements of the system. The next step in the algorithmic development phase that needed to be done was the investigation of how iBeacon data packets transmit location based information. The findings in relation to this are summarized below [18].

4.4 IBEACON

iBeacon is Apple's Trademark for location and proximity detection technology. iBeacon technology relies on Bluetooth LE as its underlying communication protocol. This satisfies the requirement 1 of the list presented above. Mobile devices are able to detect when an iBeacon is nearby, and, therefore, Mobile Apps can listen for iBeacon signals in the physical world and react accordingly. This mechanism is used in this project for testing purposes.

This satisfies requirement 4 above. iBeacon was originally designed for iOS, however a SDK has been released for Android enabling the use of this technology in Android applications.

This satisfies requirement 5 above, with the additional advantage of allowing Android support, a desirable feature for future developments of this project. A key feature of iBeacon technology is that the application can run in the background and only show up when another iBeacon is detected. iBeacon transmitters have been designed to be small and easily deployed which also fulfils requirement 2 above.

An iBeacon advertisement provides location based information via Bluetooth LE in the form of packets of data. Apple has standardized the format of Bluetooth Low Energy Advertising such that iBeacon advertisement packets contain the following pieces of information [16]:

- Proximity UUID (16 byte): a unique identifier used to distinguish a set of iBeacon transmitters from another.
- Major (2 byte): used to group a set of related iBeacon transmitters. For example, all iBeacon transmitters inside the same location will have the same Major number which will help distinguish one location from another.
- Minor (2 bytes): used to distinguish individual iBeacon transmitters within a group of iBeacon transmitters. For example, each iBeacon transmitter within a particular location will have a different Minor number in order to help distinguish which part of that location the user is in.
- Transmission Power (Tx) (2 bytes): a representation of the strength of the signal measured at one meter from the device.

The last stage of the component selection was the identification of devices that are based on iBeacon technology and are to be used within the system.

The choice of the iBeacon transmitter to be used was really only based on factors such as cost, availability, size and battery life. Kontakt beacons was identified as a suitable device to be used as the iBeacon transmitters within this project.

4.5 KONTAKT BEACONS

Kontakt beacons can be used in iOS and android which is one advantage when compared to other beacons commercially available in the market. The figure below shows the structure of the Kontakt beacons and the comparison table gives the better angle for the choice of selecting Kontakt beacons in this thesis. Also, it satisfies the requirements for this survey as discussed above in the iBeacon section.

Figure 3 - Kontakt Beacon

4.6 COMPARISON OF COMMERCIALLY AVAILABLE BEACONS

In this project, there is the necessity for comparing the different types of commercially available beacons in the market. The comparison is done with various types and it consulted with some of the important parameters, to be apply in this thesis.

The Beacons like Kontakt, Gimbal, Estimote, Bkon, Accent Systems, Bluecats are analyzed with the parameters given in table 4.

Product Name of Beacons	***Kontakt***	***Gimbal Proximity Series 21***	***Estimote***	***Bkon***	***Accent Systems iBKS 105***	***Bluecats***
Battery life	2 years	1.5 years	5 years / 2 years (location & proximity beacons)	1 year	30-40 mon/3-4 mon (depending on the TX power at 1s & 100ms interval)	3 year (approx.)
Battery	1.000 mAh CR2477 Cell. Battery is not replaceable due to sealed casing	4 standard AA alkaline batteries	–	Replaceable AAA batteries	Coin Cell CR2477 3V – 1000mAh	-20°C to +87°C (Battery Limit) / 2x AA
Range (in meters)	70m	~50m	200m/70m respectively	100+ meters at the highest power setting	70m	100m
Weight	28 grams (0.99 oz.)	6oz (170g) including batteries	–	41 grams with batteries	24g	50.5g (without batteries)
Cost	81€ / (3 beacons)	30$ per beacon	99€ / 59€ (3 beacons)	30$ per beacon	50€ / (3 beacons)	87$ (3 beacons)

Table 5 – Comparison of commercially available beacons

By the analysis of different types of commercially available beacons, Kontakt.io beacon is chosen for this project in order to its better specifications in comparison to all other types. Kontakt.io beacons are available in different models, different use cases. From these model, the “Simple Beacon” was chosen for the experiment. This beacon is analyzed by its battery power, price and distance covering ability (range) [17].

Specifications	***Gateway Beacon*** 	***Card Beacon*** 	***USB Beacons*** 
Battery life	USB Powered	Battery life about ***8 months*** (without power the saving feature), > ***1 year*** *with power saving*	Lifetime
Battery	Micro USB port	Lithium Manganese Dioxide Battery - capacity 320 mAh	USB Socket (5V power supply)
Range (in meters)	***0 to 50m***	***0 to 10m (or) 0 to 50m***	***0 to 70m***
Weight	114 grams (4.02 oz)	20 grams (0.70 oz)	4.5 grams (0.16 oz)
Cost	***$89 (1 Beacon)***	***$87 (3 Beacons)***	***$60 (3 Beacons)***
Special Features	***Real-Time Bluetooth Scanning,*** LED, Mounting Clip	***On/Off Switch,*** 2 mm thin, BLE+RFID+NFC, Customizable	Desktop Configuration App

Specifications	*Simple Beacon* 	*Beacon Pro* 	*Tough Beacon* 
Battery life	Kontakt.io profile ***(625ms** interval)* Up to ***4 years*** with default setting & 24-hours daily usage. iBeacon profile ***(100ms** interval)* Up to ***12 months*** with full & 24-hours daily usage.	Capacity of *1000 mAh* each - battery powered device lifetime up to ***60 months*** Independent supply through micro USB socket from 5V DC (standard USB voltage)2	Kontakt.io profile ***(350ms** interval)* Up to ***2 years*** with default setting & 24-hours daily usage. iBeacon profile ***(100ms** interval)*Up to ***6 months*** with full & 24-hours daily usage
Battery	2 x 1.000mAh CR2477 ***Cell, replaceable***	Three CR2477 Lithium Manganese Dioxide Coin Battery	1.000mAh CR2477 ***Cell type, Battery is not replaceable*** due to sealed casing
Range (in meters)	***0 to 70m***	***0 to 80m***	***0 to 70m***
Weight	35 grams (1.23 oz)	71 grams (2.5 oz)	28 grams (0.99 oz)
Cost	***$60 (3 Beacons)***	***$87 (3 Beacons)***	***$81 (3 Beacons)***
Special Features	-	***Real-Time Clock, USB version available,*** LED, Mounting Clip, Water-Proof	***Mounting bracket***

Table 6 – Comparison between kontakt.io beacons

5. PROJECT

5.1 COMPUTATIONAL SYSTEM

The proposed algorithms must be able to be executed in a Raspberry Pi board. This board was chosen by the company given the good compromise between price, performance, consumption and physical dimensions. This board will be installed in the drone and powered by its power supply.

Raspberry Pi is a credit-card sized computer manufactured and designed in the United Kingdom by the Raspberry. Pi foundation with the intention of teaching basic computer science to school students and every other person interested in computer hardware, programming and DIY-Do-it Yourself projects [24].

The Raspberry Pi has a Broadcom BCM2835 system on a chip (SoC), which includes an ARM1176JZF-S 700 MHz processor, VideoCore IV GPU and was originally shipped with 256 megabytes of RAM, later upgraded (Model B & Model B+) to 512 MB. It does not include a build-in hard disk or solid-state drive, but it uses an SD card for booting and persistent storage, with the Model B+ using MicroSD [26].

The foundation provides Debian and Arch Linux ARM distributions for download. Tools are available for Python as the main programming language, with support for BBC BASIC (via the RISC OS image or the Brandy Basic clone for Linux), C, Java and Perl.

Figure 4- Raspberry pi

5.2 STRUCTURE OF THE PROPOSED SYSTEM

In the previous chapter, the comparison of different beacons and the selection of beacons were discussed. It is well known that RSSI from the beacon is strongly correlated to the distance between the device and the beacon. However, the RSSI is affected by noise which is essentially fluctuation in Bluetooth signals. In order to decrease the effect of these fluctuations, a Kalman filter is employed in the proposed solution.

The structure of the proposed solution is illustrated in Figure 5. In this model of operation, the beacons broadcast Bluetooth signal. The software on the Raspberry Pi reads the RSSI values associated to the receptions from each beacon. These data are fed into a Kalman filter for reducing the impact of the noise contained in the RSSI measurements. After the filtering process, the filtered RSSI values are used to find the current position of the drone by trilateration.

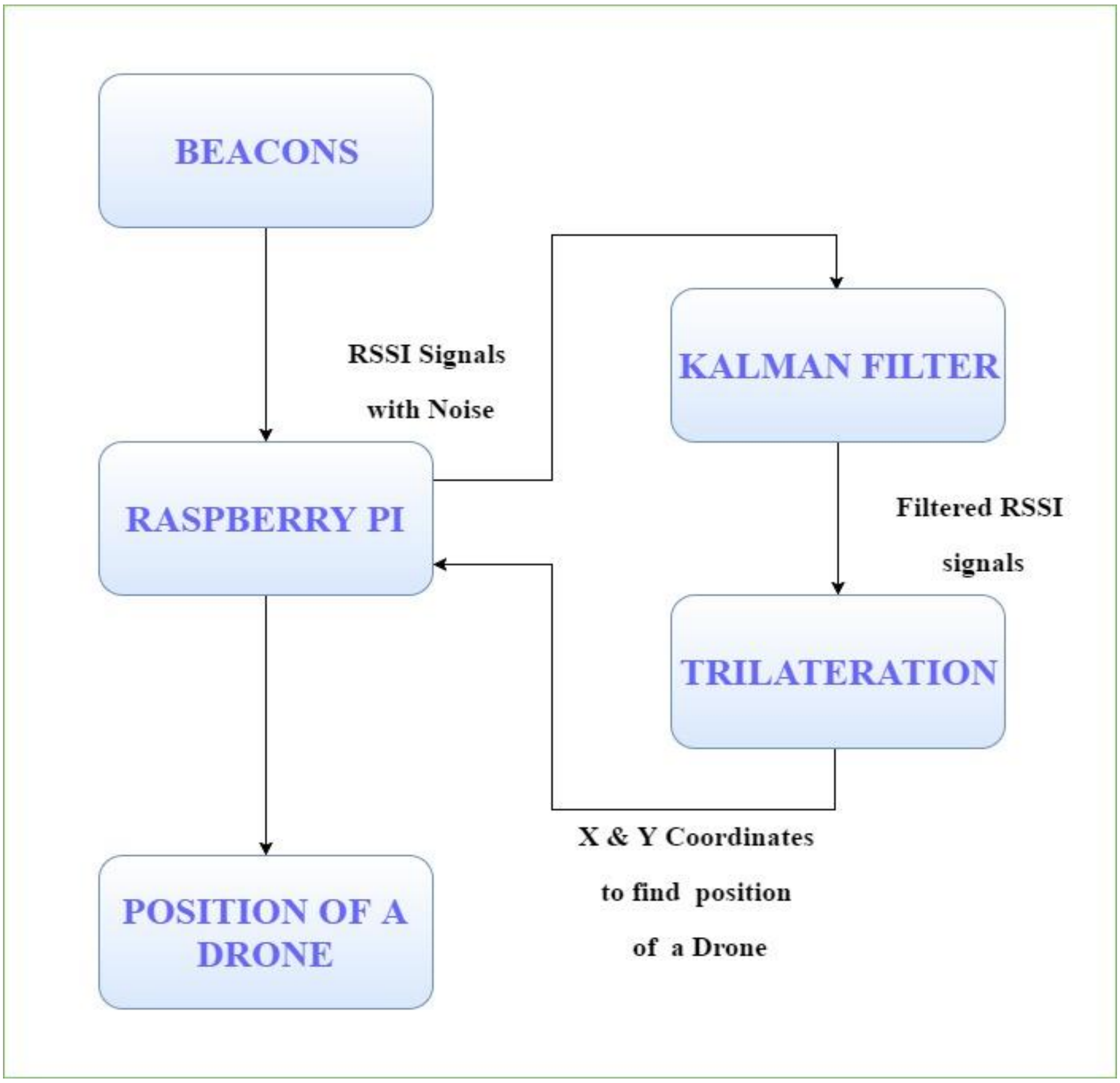


Figure 5 - Basic Structure of the Proposed System

In a first phase, the measurement error from each beacon is characterized by means of a set of experimental tests. Then, the collected data is used to design a simple scalar Kalman filter for each beacon, in order to get the best estimate from the noisy measurements.

5.3 EXPERIMENTAL SETUP

The test setup involved three beacons placed as illustrated in Figure 6. The beacons were deployed in the walls evenly, at a height of 3 m from the ground (see Figure 7). Coordinates of beacons 1, 2, and 3 are (1, 0), (1, 5.2), (5.2, 2.6) respectively (all distances in meters). The test points are labelled from A to I. Measurements from the device to each test point were then performed. To study the relation between RSSI and the real distance, the average RSSI from all the points was computed and converted to distance (in meters).

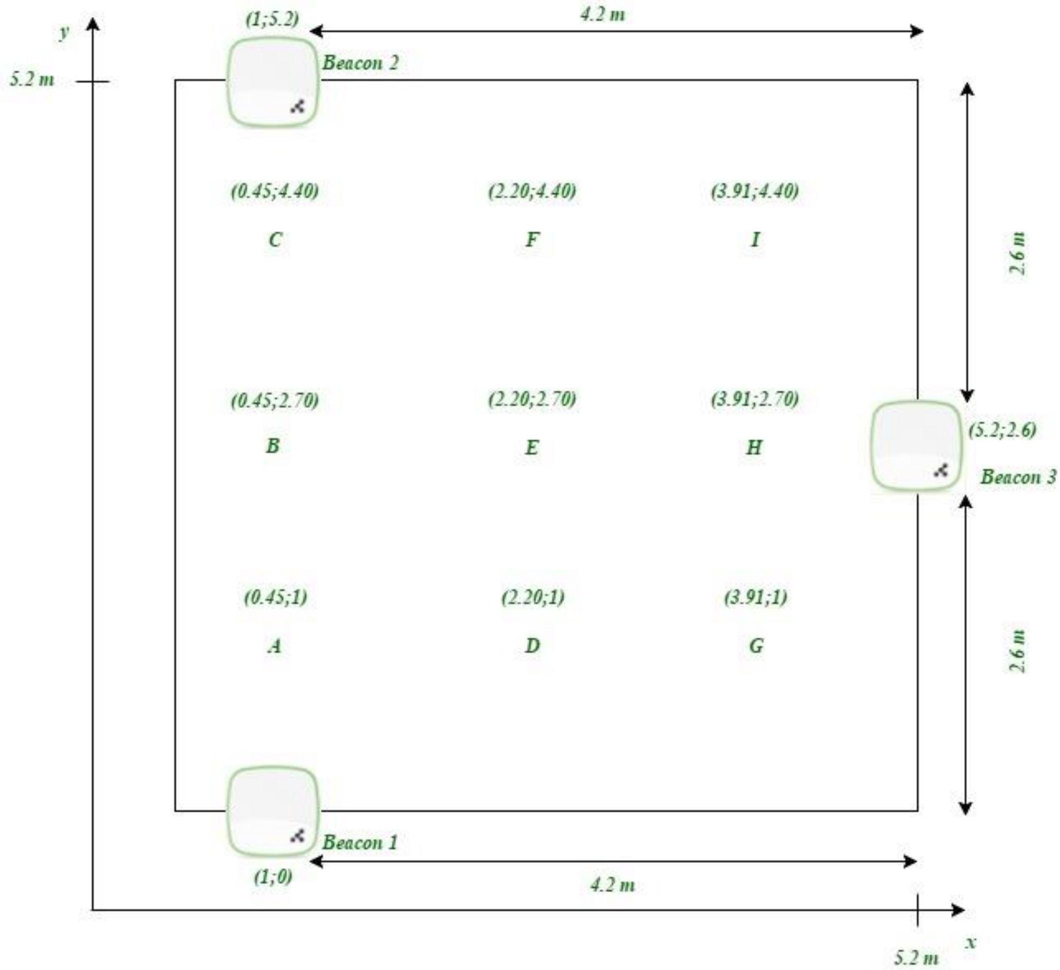


Figure 6 - Distribution of the Beacons and the Mesh points

The beacons were set to the lowest advertising rate of once per second and the highest broadcasting power of 4 dBm.

For practical reasons, the tests were performed using the Bluetooth module of a smartphone instead of the Raspberry Pi board. The source code used during the experiments is presented on Appendix A.

Figure 7– Test room during Experiment

6. TEST RESULTS AND ERROR ANALYSIS

6.1 SIMULATION TEST

For each of the nine points shown in figure 7, the position returned by the trilateration method, intentionally based on erroneous input data, was compared to the real position. This was repeated for different measurement errors and different distance measurements.

The objective of this analysis is to investigate if it is possible whether an accuracy of 0.5 m could be guaranteed if the error in distance measurements is below a certain threshold.

The result of the simulation test is summarized in Table 7 and in appendix C. The simulations show that, in order to ensure a positioning error not greater than 0.5 m, the maximum allowed error in the distance measurements is approximately 4 meters. This error was calculated for point G when the signal from the closest beacon was too long. The average error for all the 9 points is 1.25 m which is almost as low as the maximum tolerated positioning error shown in the table 8.

All test points in figure 6 were used. The results from this test can be seen in table 8. From this test the following conclusions can be drawn:

• The algorithm for trilateration is more sensitive for too short measurements than for too long measurements

• One possible improvement could be a compensation function: For the case when an intersection point cannot be found for the measured distances, the function first tries to add multiples of 0.2 meter to the distance measurements until an intersection point can be found. The algorithm remembers what distances have been added and then it repeats the same procedure but instead of adding 0.2 meters to the measurements it subtracts the 0.2 meters and then compares the total added and subtracted distances. The one with the smallest absolute value is probably the one that gives the position closest to the real position [28].

6.2 MEASUREMENT TESTS

LOCATE is an android application that, can be used for finding the distance from each beacon inside the room. It shows the distance, RSSI, major, minor and Mac address of each beacon. Using this application from the mobile the testing was done for all the nine points inside the room.

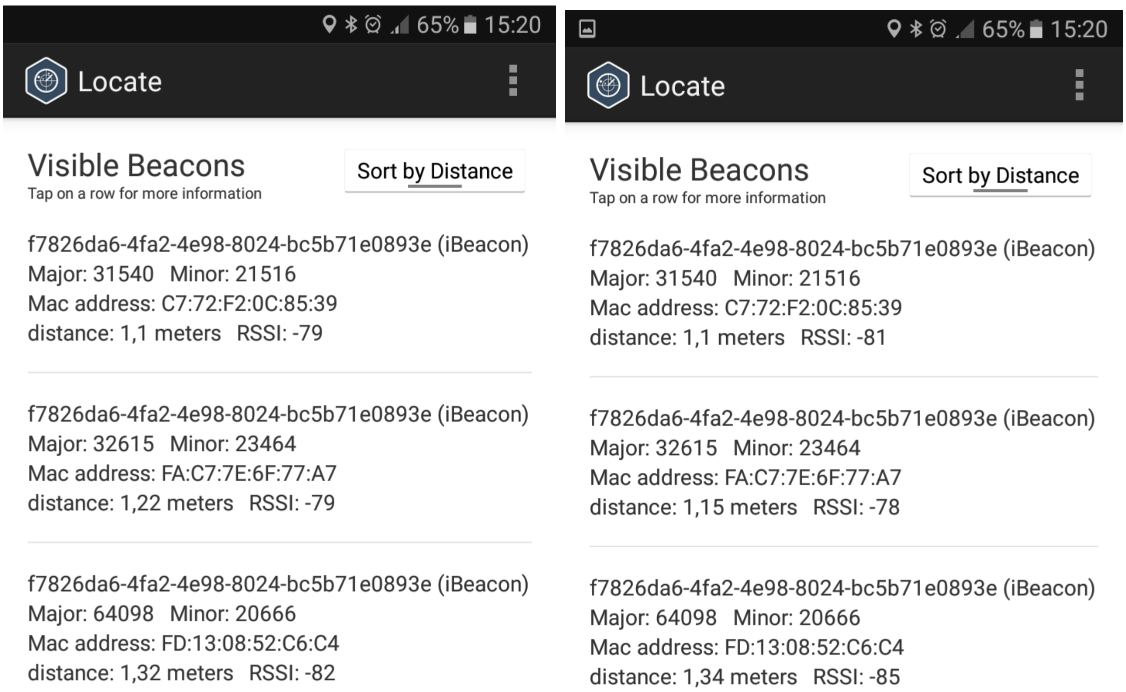


Figure 8 - Locate App Readings from Beacons

The gathered data is presented in appendix C. Table 1 presents a summary of the results.

Beacon (B1,B2,B3)	Location (A,...,I)	Average measurement error (m)	Real distance (m)	Average error (m)	Standard deviation (m)
B1	A	0.117654	1.141271	-1.06663	2.425171
B1	B	-1.17056	2.755449		
B1	C	-1.61586	4.434242		
B1	D	-0.78326	1.414214		
B1	E	-0.06085	2.879236		
B1	F	-3.39019	4.512206		
B1	G	-2.35963	3.067572		
B1	H	-3.16799	3.962323		
B1	I	-4.37847	5.269725		
B2	A	-2.97693	4.235859	B2 0.101065	B2 3.209944
B2	B	-1.43777	2.559785		
B2	C	1.847559	0.970824		
B2	D	-3.75507	4.317407		
B2	E	-1.80133	2.692582		
B2	F	-0.83394	1.280625		
B2	G	-3.51903	5.10392		
B2	H	-3.2665	3.828838		
B2	I	-2.30038	3.008322		
B3	A	-3.23396	5.012235	B3 0.427604	B3 2.883248
B3	B	-2.23917	4.751053		
B3	C	-2.84089	5.079616		
B3	D	-2.31878	3.577709		
B3	E	0.77951	3.201562		
B3	F	-0.85313	3.671512		
B3	G	-0.28327	2.061553		
B3	H	0.108697	1.30384		
B3	I	1.327774	2.22036		

Table 7- Standard deviation and average error

Based on the results from the measurements tests, namely on the standard deviation of the distance measurements, the value of Q, in the Kalman filter, was set to 10 m^2. Assuming the drone is supposed to move slowly and smoothly, the value of R was set to 1 m^2 [2,7].

From Figure 9 to Figure 14, it is possible to see the difference between the data before and after applying the Kalman filter. Each graph presents the estimated distance from every test position to a given beacon at different instants (reading number). The graphs clearly show that there are huge fluctuations in the results from the beacons before and even after applying filter.

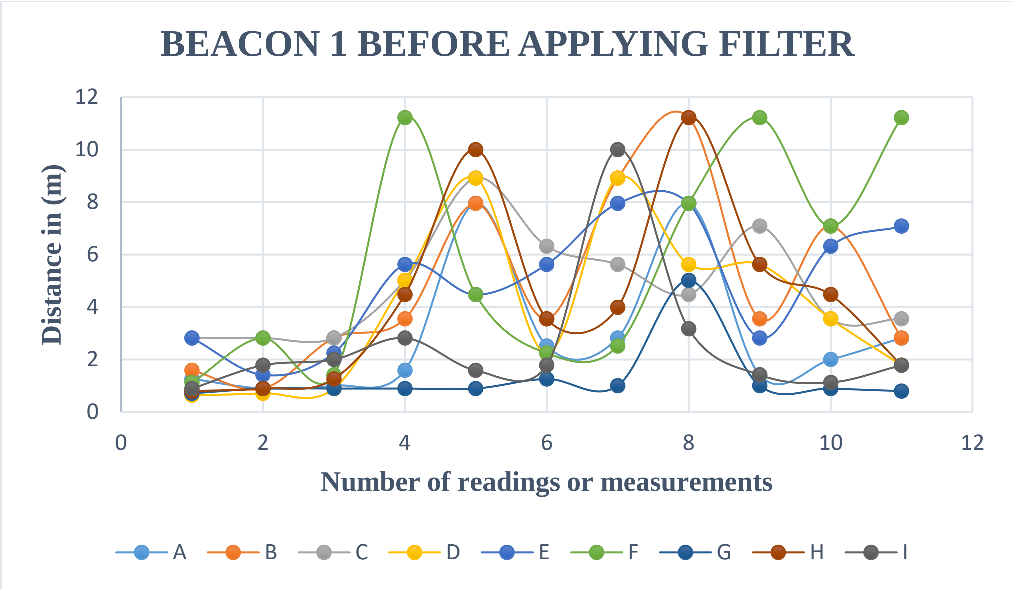


Figure 9 – Beacon 1 before applying filter

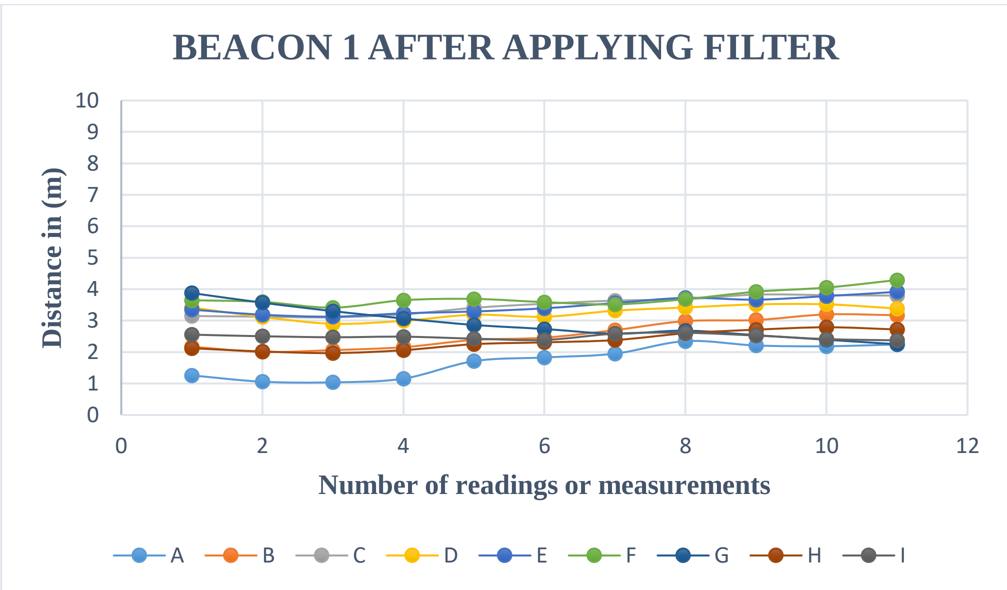


Figure 10– Beacon 1 after applying filter

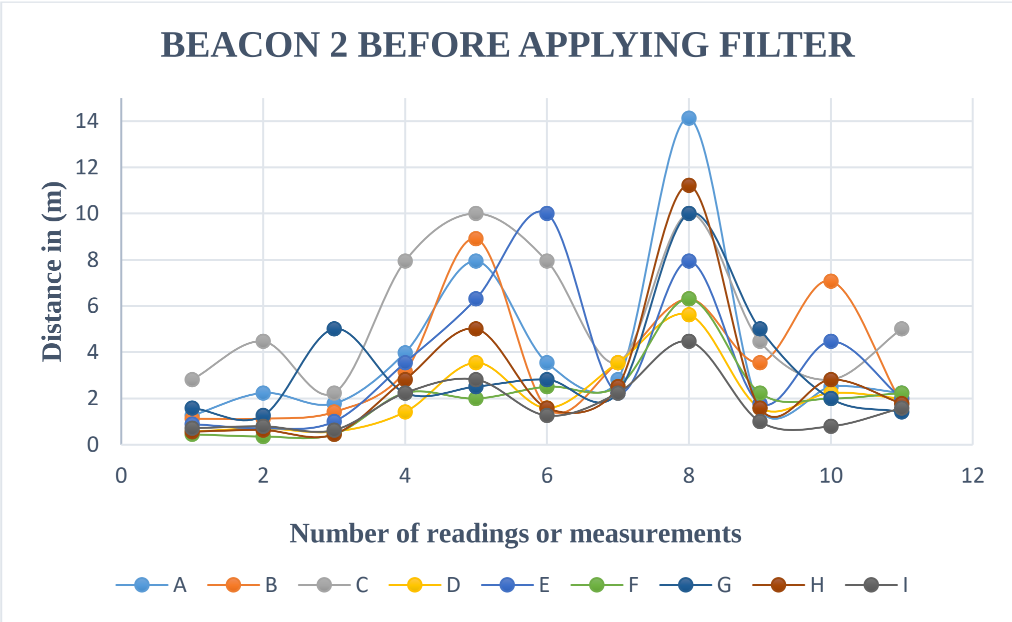


Figure 11 – Beacon 2 before applying filter

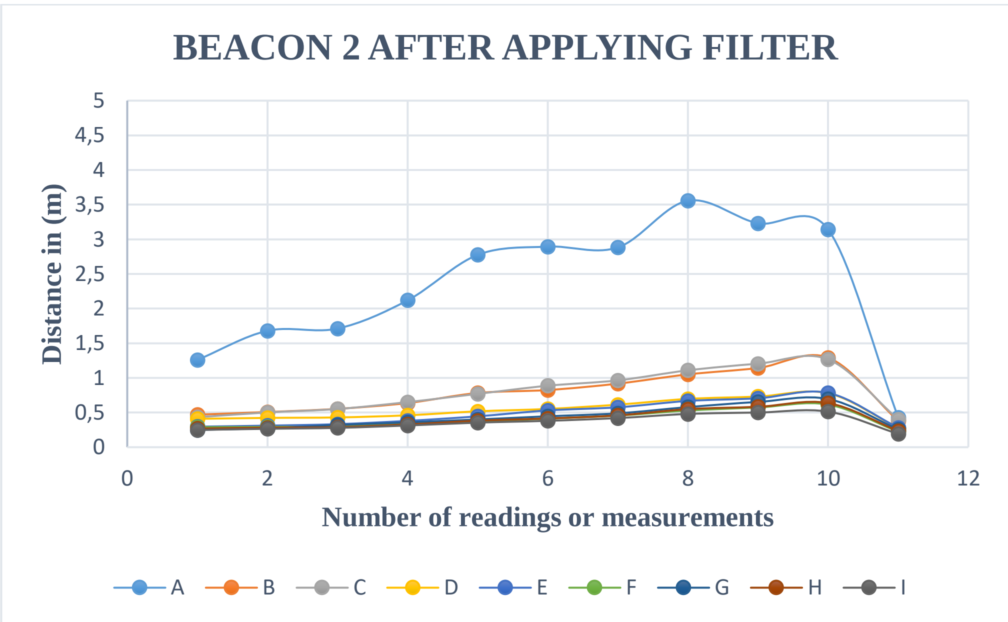


Figure 12 – Beacon 2 after applying filter

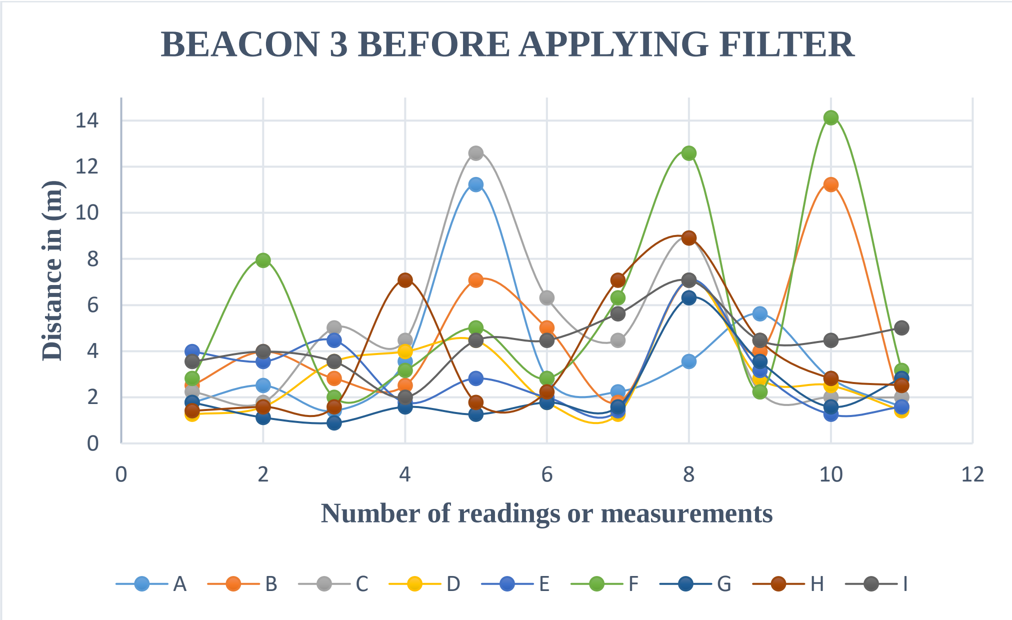


Figure 13 – Beacon 3 before applying filter

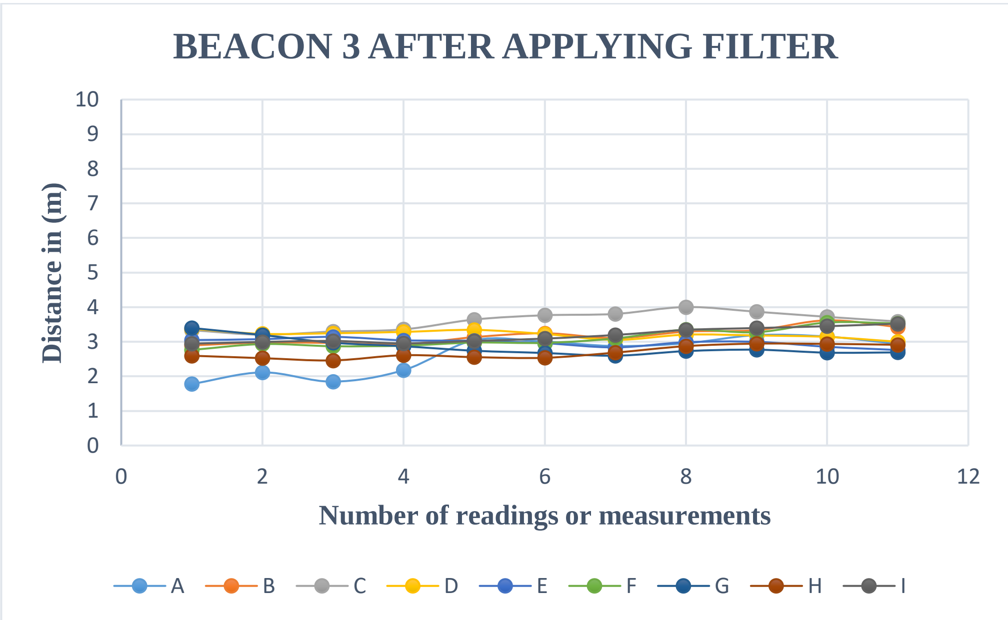


Figure 14 – Beacon 3 after applying filter

Table 8 shows the positioning error obtained by trilateration for each of the test point as also the average error of the 9 computed positions.

Trilateration		Real values		Error in X and Y		Average Error (m)
X (m)	**Y (m)**	**X (m)**	**Y (m)**	**X (m)**	**Y (m)**	
2,107	2,6	0.45	1	1.657	1.6	**X= 1,257889** **Y= 1,282667**
1,591	3,031	0.45	2.7	1.141	0.331	
1,573	3,534	0.45	4.4	1.123	0.866	
1,646	3,706	2.2	1	0.554	2.706	
1,862	3,67	2.2	2.7	0.338	0.97	
2,183	3,872	2.2	4.4	0.017	0.528	
1,819	4,039	3.91	1	2.091	3.039	
1,765	3,025	3.91	2.7	2.145	0.325	
1,655	3,221	3.91	4.4	2.255	1.179	

Table 8 – Average Error Calculation

7. CONCLUSION AND FUTURE WORK

7.1 SUMMARY

In this project the idea was to begin with beacon technology (Bluetooth low Energy), and by receiving RSSI data from beacons, to apply a filter to the RSSI data and finally use the trilateration method to find the position of an object (mobile phone or Raspberry pi). The filtering was performed using a Kalman filter.

The purpose of the experimental tests was to test the accuracy of trilateration for a small environment, and the ability to pinpoint the drone or Bluetooth device position This test was used to see how accurate the location of an object (Bluetooth device) could be found using by trilateration of the estimated ranges to the Bluetooth beacons.

It was observed that the RSSI signal noise level obtained with the smartphone is high. Even with the Kalman filter, errors of approximately 2 m were observed, which is not enough to allow safe indoors flight of a drone. This large error value may be partially explained by the fact that the Bluetooth device was being held by the candidate.

Therefore, the results obtained with the proposed system, using the Kontakt beacons, do not meet the requirements presented by the company. There is still room to improve this system's performance like deploying more BLE beacons and more elaborate filtering algorithms. The low cost of the Kontakt beacons mean the system can be cost effective, and would be a solid foundation for other low cost applications to build upon.

7.2 FUTURE WORK

To keep the prototype simple and to meet the budget of the project allowed by the company, only three beacons were used. The scalability of the system needs to be more thoroughly investigated in the future. This testing should involve more beacons (or a higher density of beacons) and a larger indoor area with more beacons. So, it would be interesting to see the performance with obstacle's in future [13].

Theoretical comparisons of existing positioning systems using WiFi may not be accurate. Therefore, a practical comparison between the proposed system and a system using WiFi can be performed in future to understand the differences in performance.

By analyzing different products with better accuracy, "Marvelmind Robotics" is one of the possible solution in future because the marginal error is just 2 cm at the cost of 400 USD. Another option is Pozyx kit, which gives 10 cm error at the cost of approximately 250 Euro.

Path loss model needs to be further studied in the future. In this project, distance model has been used, presenting good results, but it can be improved. It would need a complete statistical study of RSSI of BLE and a reconfiguration in the way data is captured and processed.

In order to have an IPS that could be adapted depending on situation, different calibration methods must be implemented in the application in order to change its configuration depending on its purpose.

BIBLIOGRAPHY

[1] Alfred Zimmermann, Alexander Rossmann (Hrsg.): "*Feasibility of Bluetooth iBeacons for Indoor Localization*", Digital Enterprise Computing 2015, Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2015

[2] Bulten.W, "*Human SLAM Simultaneous Localisation and Configuration (SLAC) of indoor Wireless Sensor Networks and their users*" September 24, 2015

[3] Song Chai, Renbo An and Zhengzhong "*An Indoor Positioning Algorithm Using Bluetooth Low Energy RSSI*" Du College of Electrical & Information Engineering, Southwest University for Nationalities, Chengdu, China

[4] R, Mardeni and Othman, Shaifull Nizam Faculty of Engineering, Multimedia University 63100 Cyberjaya, Selangor, Malaysia "*Node Positioning in ZigBee Network Using Trilateration Method Based on the Received Signal Strength Indicator (RSSI)*" http://www.eurojournals.com/ejsr.htm European Journal of Scientific Research ISSN 1450-216X Vol.46 No.1 (2010), pp.048-061 © EuroJournals Publishing, Inc. 2010

[5] Pablo Cotera , Miguel Velazquez , David Cruz , Luis Medina and Manuel Bandala, "*International Journal of Advanced Robotic Systems - Indoor Robot Positioning using an Enhanced Trilateration Algorithm"*, 03 August 2015; Accepted 20 March 2016

[6] Mautz, Rainer, *Indoor Positioning Technologies* - Application for Venia Legendi in Positioning and Engineering Geodesy - Habilitation Thesis submitted to ETH Zurich, February 2012

[7] Jenny Röbesaat, Peilin Zhang ,Mohamed Abdelaal and Oliver Theel, "*An Improved BLE Indoor Localization with Kalman-Based Fusion*": An Experimental Study, 26 April 2017

[8] Agustí Corbacho Salas "*Indoor Positioning System based on Bluetooth Low Energy"* A Degree's Thesis Submitted to the Faculty of the Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona Universitat Politècnica de Catalunya by Agustí Corbacho Salas, June 2014

[9] STALINBABU THUMMALAPALLI – author, "*Wi-Fi Indoor Positioning"* - Master report, IDE 1254, September 2012 Master Thesis in Information Technology, School of Information Science, Computer and Electrical Engineering Halmstad University, PO Box 823, SE-301 18 HALMSTAD, Sweden

[10] "*Indoor localisation using Bluetooth LE for event advertising"*, Projects:2015s1-25, https://www.eleceng.adelaide.edu.au/students/wiki/projects/index.php/Projects:2015s1 25_Indoor_localisation_using_Bluetooth_LE_for_event_advertising

[11] Hui Liu, Student Member, "*Survey of Wireless Indoor Positioning Techniques and Systems"* IEEE, Houshang Darabi, Member, IEEE, Pat Banerjee, and Jing Liu

[12] Yankai Wang, Qingyu Yang, Guangrui Zhang, Peng Zhang, "*Indoor Positioning System Using Euclidean Distance Correction Algorithm with Bluetooth Low Energy Beacon"* 2016 International Conference on Internet of Things and Applications (IOTA) Maharashtra Institute of Technology, Pune, India 22 Jan - 24 Jan, 2016,

[13] Di Zhai and Zihuai Lin, "*RSSI-based Indoor Positioning with Biased Estimator and Local Geographical Factor"* 22nd International Conference on Telecommunications (ICT 2015), Di Zhai and Zihuai Lin School of Electrical and Information Engineering, University of Sydney, Australia

[14] F. Lemic, J. Busch, M. Chwalisz, V. Handziski, and A. Wolisz, *"Infrastructure for benchmarking rf-based indoor localization under controlled interference,"* in *Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS), 2014*, Nov 2014.

[15] Santosh Subedi, Goo-Rak Kwon, Seokjoo Shin, Suk-seung Hwang, and Jae-Young Pyun "*Beacon Based Indoor Positioning System Using Weighted Centroid Localization Approach"*, 978-1-4673-9991-3/16/$31.00 ©2016 IEEE, Dept. of Information and Communication Engineering, Dept. of Computer Engineering, and Dept. of Electronic Engineering Chosun University Gwangju, 501-759, Korea

[16] Akihiro Fujihara and Takuma Yanagizawa ,2015 International Conference on "*Intelligent Networking and Collaborative Systems Proposing an extended iBeacon system for indoor route guidance"* Department of Management Information Science, Fukui University of Technology, 3-6-1 Gakuen Fukui 910-8505, JAPAN, 978-1-4673-7695-2/15 $31.00 © 2015 IEEE DOI 10.1109/INCoS.2015.72

[17] "*Bluetooth Beacons & Location Solutions*." https://kontakt.io/products-and-solutions/beacon-hardware-firmware/ ,

[18] Alcal´a de Henares, International Conference on "*Indoor Positioning and Indoor Navigation (IPIN)"*, 4-7 October 2016, Niranjini Rajagopal, Sindhura Chayapathy, Bruno Sinopoli, Anthony Rowe, Spain – "*Beacon Placement for Range-Based Indoor Localization"* Electrical and Computer Engineering Department Carnegie Mellon University 978-1-5090-2425-4/16/$31.00 c_ 2016 IEEE

[19] Mohd Ezanee Rusli, Mohammad Ali, Norziana Jamil, Marina Md Din "*An Improved Indoor Positioning Algorithm Based on RSSI-Trilateration technique for Internet of Things (IOT)"*, Systems and Networking Department, College of Computer Science and Information Technology Universiti Tenaga Nasional Malaysia - 2016 IEEE DOI 10.1109/ICCCE.2016.28

[20] "*Signal and Image Processing - The Study of the Indoor Positioning Based on Smart Wearable Technologies*," 2016 IEEE International conference on 978-1-5090-2377-6/16/$31.00 ©2016 IEEE

[21] “*Indoor Positioning System based on Bluetooth Low Energy”* A Degree's Thesis Submitted to the Faculty of the Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona Universitat Politècnica de Catalunya by Agustí Corbacho Salas, Barcelona, June 2014

[22] “*Improving the Accuracy of Bluetooth Low Energy Indoor Positioning System Using Kalman Filtering*”, 2016 International Conference on Computational Science and Computational Intelligence

[23] Vibhu Varshney, Rajat Kant Goel, Mohammed Abdul Qadeer , “*Indoor Positioning System Using Wi-Fi & Bluetooth Low Energy Technology”* Department of Computer Engineering, Aligarh Muslim University, Aligarh 202002, India, 978-1-4673-8975-4/16/$31.00 ©2016 IEEE

[24] The Official RASPBERRY PI PROJECTS BOOK, swap.raspberrypi.org, https://www.raspberrypi.org/magpi-issues/Projects_Book_v1.pdf

[25] https://en.wikipedia.org/wiki/Trilateration and http://www.mio.com/technology-trilateration.htm , iBeacon Insider. (2014). 'What is iBeacon? A Guide to iBeacons'. [Online]. Available:http://www.ibeacon.com/what-is-ibeacon-a-guide-to-beacons/. [Accessed: 17- Apr-2015].

[26] International Journal of “*Innovative and Emerging Research in Engineering - Raspberry Pi Technology”*: A Review, International Journal of Innovative and Emerging Research in Engineering Volume 2, Issue 3, 2015

[27] “*RSSI based Indoor Localization for Smartphone using Fixed and Mobile Wireless”* No2013 IEEE 37th Annual Computer Software and Applications Conference DOI 10.1109/COMPSAC.2013.18

[28] “*RSSI based Indoor Localization with Interference Avoidance for Wireless Sensor Networks using Anchor Node with Sector Antennas”* - IEEE WiSPNET 2016 conference.

[29] Y. Chen and A. Terzis, *“On the mechanisms and effects of calibrating rssi measurements for 802.15. 4 radios,”* in *Wireless Sensor Networks*. Springer, 2010, pp. 256–271.

[30] Ramon F. Brena, "*Evolution of Indoor Positioning Technologies: A Survey*", Journal of Sensors Volume 2017 (2017), https://doi.org/10.1155/2017/2630413

APPENDIX A

PATH LOSS

```
double getDistance(int rssi, int txPower) {
    /*
     * RSSI = TxPower - 10 * n * lg(d)
     * n = 2 (in free space)
     *
     * d = 10 ^ ((TxPower - RSSI) / (10 * n))
     */

    return Math.pow(10d, ((double) txPower - rssi) / (10 * 2));
}
```

TRILATERATION

```
package Trilateration;
public class Trilaterate {

    public static Point trilaterate2d(Point p1, Point p2, Point p3) {
        Double i, j, d, x, y, z, b;
        Point a, ex, ey, ez, p4a, p4b;

        ex = vector_divide(vector_subtract(p2, p1), norm(vector_subtract(p2, p1)));

        i = multiply(ex, vector_subtract(p3, p1));
        a = vector_subtract(vector_subtract(p3, p1), vector_multiply(ex, i));
        ey = vector_divide(a, norm(a));
        ez = vector_cross(ex, ey);
        d = norm(vector_subtract(p2, p1));
        j = multiply(ey, vector_subtract(p3, p1));

        x = (square(p1.r) - square(p2.r) + square(d)) / (2 * d);
        y = (square(p1.r) - square(p3.r) + square(i) + square(j)) / (2 * j) - (i / j) * x;

        b = square(p1.r) - square(x) - square(y);

        if (Math.abs(b) < 0.0000000001) {
            b = 0.0;
        }
```

```
        z = Math.sqrt(b);

        if (z == null) {
            return null;
        }

        a = vector_add(p1, vector_add(vector_multiply(ex, x), vector_multiply(ey, y)));

        return a;
    }

    public static Point[] trilaterate3d(Point p1, Point p2, Point p3, boolean return_middle) {
        Double i, j, d, x, y, z, b;
        Point a, ex, ey, ez, p4a, p4b;

        ex = vector_divide(vector_subtract(p2, p1), norm(vector_subtract(p2, p1)));

        i = multiply(ex, vector_subtract(p3, p1));
        a = vector_subtract(vector_subtract(p3, p1), vector_multiply(ex, i));
        ey = vector_divide(a, norm(a));
        ez = vector_cross(ex, ey);
        d = norm(vector_subtract(p2, p1));
        j = multiply(ey, vector_subtract(p3, p1));

        x = (square(p1.r) - square(p2.r) + square(d)) / (2 * d);
        y = (square(p1.r) - square(p3.r) + square(i) + square(j)) / (2 * j) - (i / j) * x;

        b = square(p1.r) - square(x) - square(y);

        if (Math.abs(b) < 0.0000000001) {
            b = 0.0;
        }

        z = Math.sqrt(b);

        if (z == null) {
            return null;
        }

        a = vector_add(p1, vector_add(vector_multiply(ex, x), vector_multiply(ey, y)));
        p4a = vector_add(a, vector_multiply(ez, z));
        p4b = vector_subtract(a, vector_multiply(ez, z));

        if (z == 0 || return_middle) {
            Point[] returnPoint = new Point[1];
            returnPoint[0] = a;
            return returnPoint;
        }
        else {
            Point[] returnPoint = new Point[2];
            returnPoint[0] = p4a;
            returnPoint[1] = p4b;
            return returnPoint;
        }
    }

    private static Double square(Double a) {
        return a * a;
    }

    private static Double norm(Point a) {
        return Math.sqrt(square(a.x) + square(a.y) + square(a.z));
    }

    private static Double multiply(Point a, Point b) {
        return a.x * b.x + a.y * b.y + a.z * b.z;
    }

    private static Point vector_subtract(Point a, Point b) {
        return new Point(a.x - b.x, a.y - b.y, a.z - b.z);
    }

    private static Point vector_add(Point a, Point b) {
        return new Point(a.x + b.x, a.y + b.y, a.z + b.z);
```

```
    }

    private static Point vector_divide(Point a, Double b) {
        return new Point(a.x / b, a.y / b, a.z / b);
    }

    private static Point vector_multiply(Point a, Double b) {
        return new Point(a.x * b, a.y * b, a.z * b);
    }

    private static Point vector_cross(Point a, Point b) {
        return new Point(a.y * b.z - a.z * b.y, a.z * b.x - a.x * b.z, a.x * b.y - a.y * b.x);
    }
}

package Trilateration;

public class Point {
    public double x;
    public double y;
    public double z;
    public double r;

    public Point() {
    }

    public Point(double x, double y, double z) {
        this.x = x;
        this.y = y;
        this.z = z;
    }

    public Point(double x, double y, double z, double r) {
        this.x = x;
        this.y = y;
        this.z = z;
        this.r = r;
    }
}
```

KALMAN FILTER

```
#pragma once
#include <tgmath.h>

class KalmanFilter {
private:
        float R;
        float Q;
        float A;
        float C;
        float B;
        float cov;
        float x;
public:
        KalmanFilter(float R, float Q, float A, float B, float C);
        float filter(float z, float u);
};

inline KalmanFilter::KalmanFilter(float R, float Q, float A, float B, float C)
{
        this->R = R;
        this->Q = Q;
        this->A = A;
        this->C = C;
        this->B = B;
        this->cov = NULL;
        this->x = NULL;
}

inline float KalmanFilter::filter(float z, float u)
{
        if (this->x == NULL) {
                this->x = (1 / this->C) * z;
```

```
            this->cov = (1 / this->C) * this->Q * (1 / this->C);
        }
        else {
            const float predX = (this->A * this->x) + (this->B * u);
            const float predCov = ((this->A * this->cov) * this->A) + this->R;
            float K = predCov * this->C * (1 / ((this->C * predCov * this->C) + this->Q));
            this->x = predX + K * (z - (this->C * predX));
            this->cov = predCov - (K * this->C * predCov);
        }

        return this->x;
}
```

APPENDIX B

The different types of Kontakt.io beacon pictures are given below and it can be used for the different purpose and needs.

Beacon	

Beacon pro	
Card Beacon	
Gateway Beacon	

USB Beacon	
Tough Beacon	

APPENDIX C

Y / X	1	2	3	4	5	6	7	8	9	10	11
1	1,258925	0,891251	1	1,584893	7,943282347	2,511886	2,818383	7,943282	1,412538	1,995262	2,818383
2	1,584893	0,891251	2,818383	3,548134	7,943282347	3,548134	8,912509	11,22018	3,548134	7,079458	2,818383
3	2,818383	2,818383	2,818383	5,011872	8,912509381	6,309573	5,623413	4,466836	7,079458	3,548134	3,548134
4	0,630957	0,707946	0,891251	5,011872	8,912509381	2,238721	8,912509	5,623413	5,623413	3,548134	1,778279
5	2,818383	1,412538	2,238721	5,623413	4,466835922	5,623413	7,943282	7,943282	2,818383	6,309573	7,079458
6	1,122018	2,818383	1,412538	11,22018	4,466835922	2,238721	2,511886	7,943282	11,22018	7,079458	11,22018
7	0,707946	0,891251	0,891251	0,891251	0,891250938	1,258925	1	5,011872	1	0,891251	0,794328
8	0,794328	0,891251	1,258925	4,466836	10	3,548134	3,981072	11,22018	5,623413	4,466836	1,778279
9	0,891251	1,778279	1,995262	2,818383	1,584893192	1,778279	10	3,162278	1,412538	1,122018	1,778279

(X and Y represents Number of Mesh point and Reading Number respectively. The data units is m)

Table 9 – Beacon 1 before applying filter

Y / X	1	2	3	4	5	6	7	8	9	10	11

1	1,258925	1,0589489	1,038807	1,155939	1,712655	1,829069	1,950765	2,348551	2,210294	2,185722	2,242694
2	2,170201	2,0045182	2,063241	2,155783	2,385723	2,457791	2,698796	2,984386	3,02016	3,19838	3,171794
3	3,147603	3,1255036	3,105239	3,198969	3,406866	3,537	3,637307	3,682265	3,828071	3,810965	3,795028
4	3,417394	3,1183869	2,900147	2,993195	3,187168	3,123201	3,316424	3,417984	3,516374	3,518155	3,384387
5	3,349423	3,1895176	3,126259	3,231767	3,291395	3,392462	3,559262	3,724046	3,666021	3,779854	3,91575
6	3,649808	3,5971206	3,412911	3,649094	3,690796	3,588558	3,517345	3,682053	3,919855	4,052145	4,290619
7	3,877796	3,5705901	3,302973	3,068845	2,86309	2,734072	2,584015	2,681884	2,537465	2,392792	2,249236
8	2,121632	2,0208776	1,967909	2,060535	2,251464	2,30967	2,381305	2,597646	2,712658	2,78962	2,720038
9	2,555024	2,5036003	2,471923	2,490176	2,427868	2,385805	2,585532	2,614899	2,526097	2,413682	2,372685

(X and Y represents Number of Mesh point and Reading Number respectively. The data units is m)

Table 10 - Beacon 1 after applying filter

Y X	1	2	3	4	5	6	7	8	9	10	11
1	1,258925	2,238721	1,778279	3,981072	7,943282347	3,548134	2,818383	14,12538	1,584893	2,511886	2,238721
2	1,122018	1,122018	1,412538	3,162278	8,912509381	1,584893	3,548134	6,309573	3,548134	7,079458	1,778279
3	2,818383	4,466836	2,238721	7,943282	10	7,943282	3,548134	10	4,466836	2,818383	5,011872
4	0,562341	0,707946	0,562341	1,412538	3,548133892	1,584893	3,548134	5,623413	1,584893	2,238721	1,995262
5	0,891251	0,707946	1	3,548134	6,309573445	10	2,238721	7,943282	1,778279	4,466836	1,995262
6	0,446684	0,354813	0,446684	2,238721	1,995262315	2,511886	2,511886	6,309573	2,238721	1,995262	2,238721
7	1,584893	1,258925	5,011872	2,238721	2,511886432	2,818383	2,238721	10	5,011872	1,995262	1,412538
8	0,562341	0,630957	0,446684	2,818383	5,011872336	1,584893	2,511886	11,22018	1,584893	2,818383	1,778279
9	0,707946	0,794328	0,630957	2,238721	2,818382931	1,258925	2,238721	4,466836	1	0,794328	1,584893

(X and Y represents Number of Mesh point and Reading Number respectively. The data units is m)

Table 11 - Beacon 2 before applying filter

Y X	1	2	3	4	5	6	7	8	9	10	11

1	1,258925	1,679616	1,712064	2,119337	2,774853	2,894476	2,883036	3,556845	3,229758	3,142281	0,42879
2	0,469656	0,5076098	0,553592	0,637434	0,782465	0,824983	0,917181	1,050933	1,142747	1,292007	0,37963
3	0,43231	0,5017761	0,551284	0,650676	0,769618	0,887044	0,964384	1,109226	1,204925	1,266922	0,401384
4	0,409369	0,42262	0,42967	0,46023	0,517631	0,551963	0,613995	0,696795	0,730214	0,778278	0,277204
5	0,296193	0,311186	0,332441	0,380067	0,445446	0,530976	0,575877	0,667698	0,705594	0,782871	0,281397
6	0,288809	0,2921729	0,29923	0,335077	0,37042	0,412487	0,456552	0,529121	0,573773	0,615368	0,223344
7	0,249316	0,2730423	0,321492	0,358488	0,39988	0,446195	0,488467	0,578642	0,653161	0,695401	0,257416
8	0,268952	0,2821305	0,289498	0,328924	0,38323	0,414997	0,459108	0,549269	0,582908	0,636788	0,233894
9	0,248886	0,2656287	0,278837	0,313397	0,354491	0,38061	0,420383	0,479977	0,50015	0,513299	0,192077

(X and Y represents Number of Mesh point and Reading Number respectively. The data units is m)

Table 12 - Beacon 2 after applying filter

Y / X	1	2	3	4	5	6	7	8	9	10	11
1	1,778279	2,511886	1,412538	3,548134	11,22018454	2,818383	2,238721	3,548134	5,623413	2,818383	1,584893
2	2,511886	3,981072	2,818383	2,511886	7,079457844	5,011872	1,778279	7,079458	3,981072	11,22018	1,584893
3	2,238721	1,778279	5,011872	4,466836	12,58925412	6,309573	4,466836	8,912509	2,238721	1,995262	1,995262
4	1,258925	1,584893	3,548134	3,981072	4,466835922	1,778279	1,258925	7,079458	2,818383	2,511886	1,412538
5	3,981072	3,548134	4,466836	1,778279	2,818382931	1,995262	1,412538	7,079458	3,162278	1,258925	1,584893
6	2,818383	7,943282	1,995262	3,162278	5,011872336	2,818383	6,309573	12,58925	2,238721	14,12538	3,162278
7	1,778279	1,122018	0,891251	1,584893	1,258925412	1,778279	1,584893	6,309573	3,548134	1,584893	2,818383
8	1,412538	1,584893	1,584893	7,079458	1,77827941	2,238721	7,079458	8,912509	4,466836	2,818383	2,511886
9	3,548134	3,981072	3,548134	1,995262	4,466835922	4,466836	5,623413	7,079458	4,466836	4,466836	5,011872

(X and Y represents Number of Mesh point and Reading Number respectively. The data units is m)

Table 13 - Beacon 3 before applying filter

(X and Y represents Number of Mesh point and Reading Number respectively. The data units is m)

Y / X	1	2	3	4	5	6	7	8	9	10	11
1	1,778279	2,1140974	1,846822	2,178437	3,043267	3,003413	2,874783	2,955868	3,191648	3,148582	2,937278
2	2,89411	2,977625	2,963808	2,924388	3,132384	3,244741	3,106061	3,292077	3,335416	3,619178	3,427283
3	3,333957	3,202912	3,294428	3,357337	3,641372	3,765174	3,804171	4,002995	3,867141	3,718947	3,585461
4	3,372795	3,2278623	3,2456	3,28409	3,342758	3,223851	3,054921	3,205125	3,181706	3,139207	2,999957
5	3,048527	3,0748575	3,140509	3,04113	3,028099	2,957604	2,836841	2,98693	2,996574	2,853711	2,7608
6	2,764012	2,9331211	2,870252	2,885925	2,976871	2,9677	3,096171	3,350001	3,275028	3,555167	3,531872
7	3,398404	3,1934123	2,972658	2,869558	2,73987	2,674207	2,596839	2,729481	2,76997	2,684541	2,691876
8	2,596211	2,5253118	2,460169	2,610478	2,554848	2,535975	2,686334	2,873294	2,945303	2,938022	2,912293
9	2,944761	2,9949874	3,023605	2,953895	3,023222	3,090153	3,195693	3,341527	3,39637	3,448975	3,522046

Table 14 - Beacon 3 after applying filter