FEDERAL UNIVERSITY OF AMAZONAS - UFAM

INSTITUTE OF COMPUTING - ICOMP

POST-GRADUATE PROGRAM IN INFORMATICS - PPGI

Robust RSSI-based Indoor Positioning System using K-Means Clustering and Bayesian Estimation

Bráulio Henrique Orion Uchôa Veloso Pinto

Manaus - AM

July 2021

Bráulio Henrique Orion Uchôa Veloso Pinto

Robust RSSI-based Indoor Positioning System using K-Means Clustering and Bayesian Estimation

A master thesis submitted to the Post-graduate Program in Informatics of the Institute of Computing of the Federal University of Amazonas in partial fulfillment of requirements for the degree of Master of Science. Concentration area: Informatics.

Advisor

Horácio A. B. Fernandes de Oliveira, D.S.c

FEDERAL UNIVERSITY OF AMAZONAS - UFAM

INSTITUTE OF COMPUTING - ICOMP

Manaus - AM

July 2021

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

Pinto, Bráulio Henrique Orion Uchôa Veloso
P659r Robust RSSI-based indoor positioning system using k-means clustering and Bayesian estimation / Bráulio Henrique Orion Uchôa Veloso Pinto . 2021
52 f.: il. color; 31 cm.

Orientador: Horácio Antonio Braga Fernandes de Oliveira
Dissertação (Mestrado em Informática) - Universidade Federal do Amazonas.

1. Bayesian estimation. 2. Indoor positioning. 3. K-means clustering. 4. Log-distance path loss model. 5. Rssi. I. Oliveira, Horácio Antonio Braga Fernandes de. II. Universidade Federal do Amazonas III. Título

FOLHA DE APROVAÇÃO

"Sistema Robusto de Localização Interna usando Agrupamento K-Means e Estimativa Bayesiana"

BRÁULIO HENRIQUE ÓRION UCHÔA VELOSO PINTO

Dissertação de Mestrado defendida e aprovada pela banca examinadora constituída pelos Professores:

Prof. Horácio Antônio Braga Fernandes de Oliveira - PRESIDENTE

Digitally signed by Richard W. Pazzi
Date: 2021.07.25 18:47:57 -04'00'

Prof. Richard Werner Pazzi - MEMBRO EXTERNO

Prof. Raimundo da Silva Barreto - MEMBRO INTERNO

Manaus, 22 de Julho de 2021

Acknowledgements

First of all, I would like to thank God for the daily graces I have been given, and the Holy Catholic Church for the teachings and references of worthy lives.

I would like to thank my mother Lolita for all the support and sacrifice, and my wife Sue Ann for being such a shining light in my life.

I want to thank the Federal University of Amazonas, represented by the Institute of Computing, for all the support and infrastructure provided for my research. A special thanks to my advisor prof. Dr. Horácio Fernandes for all his precious time spent with me in our weekly meetings, his guidance, and his ability to extract the best of me.

I would also like to thank the support given by the Foundation for Research Support of the State of Amazonas (FAPEAM), by the Coordination for the Improvement of Higher Education Personnel (CAPES), and by Samsung Electronics of Amazonia Ltda, through the agreement no 003, signed with ICOMP / UFAM, that allowed this work to be carried out.

And last but not least, I want to thank all my research colleagues for their partnership and valuable tips, and to all my friends who somewhat helped me achieve my goals.

Robust RSSI-based Indoor Positioning System using K-Means Clustering and Bayesian Estimation

Author: Bráulio Henrique Orion Uchôa Veloso Pinto

Advisor: Horácio A. B. Fernandes de Oliveira, D.S.c

Abstract

This work proposes a new indoor positioning system, named KLIP, that uses the K-means clustering algorithm to split the environment into different sets of log-distance propagation models in order to better characterize the indoor environment and further improve the position estimation using Bayesian inference. The proposed method is validated in a large-scale, real-world scenario composed of Bluetooth Low Energy (BLE)-based devices. It is demonstrated, throughout the work, that the addition of location information of training points to the received signal strength indicator (RSSI) as an attribute for the clustering step improves the positioning accuracy. Moreover, the obtained results show that the solution outperforms the naive Bayesian estimation up to 12% – regarding the positioning accuracy – and the broadly deployed kNN for reduced training dataset size – regarding both accuracy and online processing time. In this sense, KLIP proves to be an efficient and scalable alternative when both site-survey effort and energy consumption constraints must be taken into account.

Keywords: Bayesian estimation, Indoor Positioning, K-Means clustering, Log-distance Path Loss Model, RSSI.

List of Figures

List of Tables

List of abbreviations and acronymns

AP Access Point

BLE Bluetooth Low Energy

GNSS Global Navigation Satellite Systems

GPS Global Positioning System

IoT Internet of Things

IPS Indoor Positioning System

KLIP K-means- and Log-distance model-based Indoor Positioning

kNN k-Nearest-Neighbor

NLOS Non-Line-Of-Sight

RMSE Root Mean Square Error

RP Reference Point

RSSI Received Signal Strength Indicator

TB Traditional Bayesian

TP Training Point

Contents

1 Introduction

1.1 Context

Indoor Positioning Systems (IPSs) are a reality and provide location information of devices and persons for different applications in the real world. With the appropriate technology, it is possible to locate products in a warehouse, firefighters in a burning building, medicines in a hospital, maintenance tools spread over a plant, and so forth (Liu et al., 2007). Moreover, with the ascending global need for smart devices and connected networks, indoor positioning becomes one of the principal enabling technologies for a great variety of services in the context of the Internet of Things (IoT) (Macagnano et al., 2014).

Applications already well established as Google Maps, Waze, and Uber are also location-based services, except that they are used outdoors. In this case, the most widespread technology is the Global Navigation Satellite Systems (GNSS), which includes the Global Positioning System (GPS). Unfortunately, GNSS does not perform well indoors, as it needs, among other factors, direct line of sight to the satellites and the device whose location one wants to know (Soares Lima et al., 2018).

An indoor positioning system must take into account some factors whose effects compromise the accuracy when estimating the location. Lack of line of sight, the influence of obstacles and obstructions such as walls and human movement, multipath propagation, and interference noises are examples of factors that result in the low performance of the most commonly deployed solutions (Farid et al., 2013).

The technologies deployed in indoor positioning systems are often based on scenario image processing, infrared, WiFi, ultra-wideband (UWB), Bluetooth, inertial navigation, and magnetic solutions (Liu et al., 2020). However, most of IPSs use wireless

technologies such as WiFi or Bluetooth due to a wide available and accessible infrastructure, which saves time and related costs of deployment (Gu et al., 2009). A common architecture consists of mobile devices, access points (APs), and a central server. The main goal is to obtain location information of the mobile devices. To do this, the devices should transmit signals, whose power levels are captured by the access points which are spread over the environment. The power levels, well-known in the literature as Received Signal Strength Indicator (RSSI), are passed on to the central server. After that, these data are processed using techniques and appropriate algorithms to determine the location of the devices.

There is a vast literature of positioning algorithms used for IPSs, which include deterministic and probabilistic methods. The first ones are quite common in fingerprinting-based localization, which basically consists of two main steps: an offline phase, in which RSSI measurements (fingerprints) are previously collected in the environment; and an online phase, in which machine learning techniques and algorithms are used for the location estimation by a comparison between the offline database and the RSSI data collected in real time. One of the first and most traditional systems is the RADAR (Bahl and Padmanabhan, 2000), which achieved a median error of 2 – 3 m. The second, probabilistic methods, are much more common in propagation model-based systems that take into account the random component inherent to the variability of RSSI over the environment. In this case, the employed model is more likely to describe the indoor area reasonably. One advantage of this method is a better computational efficiency. A well-known probabilistic-based solution is the HORUS (Youssef and Agrawala, 2005), which achieved an error of approximately 2 m during 95% of the time for its particular testbed. Besides that, many systems provide hybrid solutions taking into account specificities of the indoor environment, seeking in general to improve accuracy. The proposed work belongs to second category, that is, a model-based IPS that benefits from the Bayes probabilistic theory for the position estimation.

1.2 Problem

Although fingerprint-based systems are usually accurate, the offline or training phase is very labor-intensive and time-consuming, since it demands a significant amount of time to gather reliable and enough RSSI samples for every selected point in large, indoor scenarios (Liu et al., 2020). Also, the computational load in the online phase is an expensive deployment factor for such systems.

To partially overcome the problem of intensive site-survey effort, some solutions rely on the log-distance path loss models, which describe, on average, the signal propagation throughout an indoor environment (Rappaport, 2002). Geometrical approaches, such as trilateration and multilateration techniques, are very efficient in this regard but have lower accuracy (Subedi and Pyun, 2020; Zafari et al., 2019). Probability-based approaches, in turn, are usually more accurate than geometrical ones, but they are not as precise as fingerprint methods for larger and more complex environments (Man et al., 2020).

The problem of computational efficiency in fingerprint-based IPSs is usually overcome by employing clustering techniques. However, the reduction in the computational load often comes with an accuracy drawback (Torres-Sospedra et al., 2020b). The advances of the area yielded different approaches concerning traditional clustering — as K-means (Altintas and Serif, 2011; Torres-Sospedra et al., 2020a; Zhong et al., 2016) —, hierarchical (Li et al., 2021; Zhang et al., 2020) and novel clustering methods (Alraih et al., 2017; Liu et al., 2016; Ren et al., 2019), and dataset compression techniques (Klus et al., 2020). These clustering techniques are often associated with algorithms based on instances, and the k-Nearest-Neighbor (kNN) is one the most deployed due to its low complexity of implementation and accuracy results above the average regarding solutions for IPSs. The main working principle of kNN is the exhaustive search over the fingerprint database for the most similar fingerprints to the current RSSI one desires to locate. In this sense, the strength of being able to find the most probable candidate location is also the weakness of spending a great amount of time to estimate the position. Particularly, the K-means clustering algorithm is often employed due to its flexibility and efficiency in finding optimized clusters. By dividing a fingerprint database into

sets of similar instances, the online phase is benefited from a reduction in the search space, i.e, the estimation takes into account the search for the appropriate cluster, which is proportional to K, and the search over the instances of that specific assigned cluster, which is expected to be drastically reduced in comparison with the whole dataset. Still, the effort put into the training phase remains relatively high to build the fingerprint database and provide acceptable accuracy results.

As one can see, the efficiency in both the site-survey effort and the online computational load is essential for the feasibility of IPSs regarding time spent on training, energy consumption, and real-time applications. Naturally, many studies are concerned with continuously improving classic solutions while keeping the positioning error at competitive levels. In this sense, to build an accurate IPS that meets the requirements of reduced effort in offline training and low computational cost when estimating the position remains a challenge for the research area.

1.3 Objectives

The main objective is to develop a novel positioning algorithm that is more accurate than the traditional Bayesian approach by using clustering techniques, and scalable in terms of online processing time and site-survey effort when compared to classic fingerprint-based schemes. The scalability in processing time should be measured in terms of robustness, which is considered here as the ability of the corresponding parameter to remain practically invariable with the increase of dataset size. On the other hand, the scalability in terms of site survey effort should be measured by the amount of training points needed to achieve acceptable positioning errors. To achieve this, some specific objectives are intended:

1. Collect RSSI samples from a real-world, large scale scenario;
2. Build the fingerprint database from the collected samples;
3. Develop the K-means Clustering and the Bayesian Estimation Algorithms;

4. Adapt the K-means to the dataset of reference in order to train the K corresponding log-distance models;

    - Analyze two different attributes and verify their impact on the system accuracy.

5. Enhance the Bayesian estimation;

    - Improve the estimation by using the weighted Bayesian approach;

6. Combine the improved K-means clustering with the Bayesian estimation to reduce the average positioning error;

7. Compare the developed solution with the classic Bayesian estimation and the commonly deployed kNN under equivalent circumstances and the same validation scenarios in terms of positioning error performance.

1.4 Applicability of the Solution

The essence of the proposed work enables a high flexibility for its deployment in a diversity of applications. Indoor spaces with an expected high variability of RSSI such as hospitals, shopping malls, schools, buildings with several floors, and so on, are included, once the diversity of signal propagation is expected to be addressed by the clustering process of the proposed work in the offline phase.

Concerning the useful information to be used by the location based services, there is also flexibility. Although the metric for the positioning error employed here is delivered in meters, a room-accuracy level error is perfectly possible as well. For more general applications, however, the error in meters is one of the most common in the literature, which, by the way, includes the tracking applications.

The solution can be also applied in different wireless infrastructures. These include architectures with a central server, as commented previously, but also decentralized architectures, in which each mobile device is responsible for computing its own location. The latter are particularly sensitive to the battery-life capacity of the devices,

which can be a drawback for systems that require a high processing time for estimation. In this sense, the proposed work is feasible due to its relatively short processing time.

1.5 Structure of the Master Thesis

The rest of this master thesis is organized as follows: Chapter 2 provides the related literature regarding model-based IPSs. In Chapter 3, the fundamental theory behind the proposed method is described. Chapter 4 presents in detail the components of the system, as well as the testbed in which the validation of the solution is conducted. Chapter 5 presents the obtained preliminary results by comparing the performance of different positioning approaches. Finally, Chapter 6 draws the conclusions, and presents the expectations for the next steps of the research.

2 Related Work

Algorithms based on propagation models are usually efficient in terms of processing time and do not require a huge dataset to achieve reasonable positioning errors. In general, the most recent solutions are based on geometrical, as trilateration and multi-lateration techniques, or probabilistic approaches. They all have in common the search for improvement over existing methods.

Njima et al. (2017) proposed an enhanced probabilistic algorithm with RSSI fingerprinting, which benefited from an AP selection strategy based on information theory. They validated their proposal in a large-scale known dataset and compared their results with the classic probabilistic approach and the kNN. The AP selection consisted of reducing the computable APs for estimating the position. This allowed only a few APs with proved RSSI diversity of information to be considered, which improved the location estimation. After that, the estimation took into account an weighted average of the most probable training points in terms of similarity with the received RSSI in the online phase. The authors achieved a lower computing complexity if compared to classic approaches. Still, a high site-survey effort is needed, as the proposed approach is an enhanced probabilistic fingerprint method, and thus many training points are required in the offline phase.

Han et al. (2018) developed a novel probabilistic method to improve accuracy by mitigating the effect of Non Line-Of-Sight (NLOS) characteristics of the environment. Compared to the trilateration and the classic probabilistic algorithms, the proposed solution reached the minimum positioning error. Nonetheless, the proposed method requires high computational load due to the use of fusion algorithms. In this sense, the battery life of receiver devices are reduced, which is addressed by the authors as a major concern.

In Li et al. (2019), a probabilistic algorithm using hidden Markov chains was proposed to improve the room-level accuracy. By adopting a crowd-sourcing approach, the authors mitigate the problem of high effort in the training phase. Also, the presence of high signal diversity due to heterogeneous devices is softened by the use of a linear regression model allied with a geometric distribution of visible APs. The proposed solution applies to both static and tracking estimations. The obtained positioning errors are less than 8%. Still, due to the complex strategy deployed in the online phase, one estimation can take several seconds to occur, which is not feasible for applications out of the room-level accuracy domain.

Wu et al. (2019) proposed a weighted centroid technique based on least squares to estimate the position. The estimation was enhanced by using numerical approximations based on the Gaussian distribution of RSSI near the BLE transmitter devices. They also discuss the impact of the numbers of APs on the accuracy. The authors validate their method in two real indoor scenarios with at most 240 m^2. Although effortless training is needed, the processing time analysis of the online estimation phase was not addressed, which is expected to be relatively high, as a composition of serial steps is required to improve accuracy.

Hoang et al. (2020) developed a semi-sequential probabilistic approach to boost the accuracy performance of traditional probability-based algorithms. Mainly focused on tracking applications, the addition of the previous position estimation proved to reduce the error up to 30%. The simple short term memory addition provided by the method in relation to the classic methods is negligible and thus is not considered a major concern for the final estimation processing time. However, the works served as references were mostly fingerprint-based, which represent only a high training effort scenario.

In Li et al. (2020) , a probabilistic method was developed to learn the best positioning strategy according to a label credibility constraint. The obtained results show the proposed algorithm is accurate in complex environments at the expense of a much higher computational cost when compared to traditional approaches presented in the work.

Alfakih et al. (2020) proposed an improved Gaussian mixture model to characterize more precisely the RSSI variability. As the probability of each location is proportional to the number of mixtures, the algorithm running time is equally proportional to this parameter. Among traditional algorithms as the kNN and its variants, as well as classic probabilistic approaches, the proposed method had the best performance in terms of positioning error. Since the proposed solution is based on probabilistic-fingerprint schemes and validated in a small scenario of 110 m^2, a demanding computational load in the online phase is expected due to the addition of mixtures, as well as a high site-survey effort for large-scale environments.

Finally, Assayag et al. (2020) presented a novel multilateration-based positioning algorithm that dynamically computes the parameters of the model used to characterize the indoor scenario. By benefiting from an improved minimum least square estimation, the method is efficient in terms of computational cost. Also the model built in the offline phase only requires one training point per access point, which is a total of fourteen in the work. In this sense, little effort is spent on the training phase. The authors prove the superiority of their proposed algorithm against the traditional least square approach and they achieve a minimum of 3.0 m for the average positioning error and around 75% for the room-level accuracy. Nevertheless, the error results might not be competitive against classic Bayesian or fingerprint-based solutions.

In summary, there are several classes of positioning algorithms that seek to improve some of the following performance metrics: training effort, accuracy, and/or processing time. To achieve a balanced solution that meets these three requirements satisfactorily remains a challenge for IPSs. Table 1 provides a simplified comparison among the described works above regarding the size of the considered validation scenario, the site-survey effort, the online running time of each proposed algorithm, and the positioning error. The reference for qualitatively classifying the training effort performance into low, medium or high is the average number of training points needed in the offline phase. For fingerprint-based schemes, the training effort is classified as "high". For model-based ones, "low". Optimized systems based on fingerprinting are classified as "medium". Similarly, for the running time is the average number of

comparisons needed for estimation.

Table 1 – Comparison table among related works.

Author	**Dimension**	**Training Effort**	**Processing Time**	**Average Error**
(Njima et al., 2017)	100 x 100 m	High	Low	5.8 m
(Han et al., 2018)	54 x 5 m	Low	High	1.2 m[(1)]
(Li et al., 2019)	-	Low	High	7%[(2)]
(Wu et al., 2019)	19 x 13 m	Low	Medium	0.9 m
(Hoang et al., 2020)	21 x 16 m	High	Medium	1.0 m
(Li et al., 2020)	308 m^2	Low	High	2.6 m
(Alfakih et al., 2020)	11 x 10 m	Medium	High	1.5 m
(Assayag et al., 2020)	45 x 16 m	Low	Low	3.0 m

(1) Root Mean Square Error (RMSE).
(2) Room-Level Accuracy Error.

Unlike the previous works, the proposed solution, named KLIP (**K**-means- and **L**og-distance model-based **I**ndoor **P**ositioning), combines the simplicity of the log-distance model and the related Bayesian theory with the K-means clustering technique to better characterize the signal propagation over the indoor environment. In this sense, the proposed method enables a significant reduction in the training effort without compromising accuracy and processing time performances. To achieve this, the collected RSSI samples are clustered into different log-distance models during the offline phase, and the Bayesian theory is applied to estimate the position at the online phase.

3 Theoretical Background

This chapter presents the main theoretical aspects of the proposed IPS, which includes the fundamentals of fingerprint-based systems, the log-distance path loss model, the Bayesian estimation, and the K-means clustering algorithm.

3.1 Fundamentals of Fingerprinting

Although the proposed work is not based on the fingerprinting technique, the principles and terms behind this subject are important for a full comprehension of many methods for IPSs.

Especially for RSSI-based systems, a fingerprint is a set of values, each corresponding to a captured signal from a determined access point over a specific point in space. The fingerprint, in this sense, is literally a stamp which is expected to represent the signal strength in some particular location. Due to the high variability of RSSI in indoor spaces, a set of fingerprints is eventually collected for a particular point to better represent the stochastic feature of the RSSI.

In order to cover an entire indoor space, an intuitive approach to capture the RSSI variability is to uniformly collect fingerprints over the environment. In this sense, a set of physically located points is chosen, and, for each point, a set of fingerprints is collected. In the literature, this phase is known as the "training" phase or the "offline" phase, and each collection point for the fingerprints is known as training point (TP). A database, in this case, is built for further query in an estimation phase. The last is also known as the "online" phase, which is responsible for estimating the location by comparing a set of new fingerprints with the dataset previously stored. The most

similar set, in this way, is a strong candidate for providing the correct answer of location information. Figure 1 summarizes the fingerprint-based method.

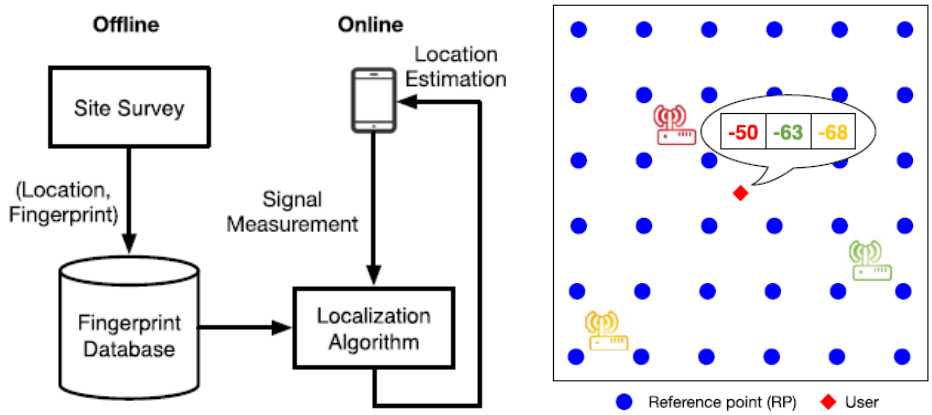

Figure 1 – Basic representation of the fingerprinting technique. Adapted from Gu et al. (2009).

Depending on the size of the indoor area, a high effort is required for the site survey process. Indeed, to collect dozens of fingerprints for each of the possible hundreds of training points is certainly a time-consuming and exhaustive task.

The main positioning algorithms that benefit from the fingerprinting technique are the machine learning- based, which include the kNN and its variations, neural networks, random forests, and so forth. Probabilistic approaches are also widely employed, in which the signal variability for each training point is described by a representative model, usually the Gaussian distribution.

3.2 Log-Distance Path Loss Model

The log-distance path loss model is empirically demonstrated to represent indoor signal propagation (Rappaport, 2002), as shown in Equation (3.1):

$$PL(dB) = PL(d_0) + 10\alpha \log\left(\frac{d}{d_0}\right) + X_\sigma \tag{3.1}$$

where $PL(d_0)$ is a constant which represents the path loss in dB at a distance d_0 used as a reference, α is the path loss exponent, and X_σ is a normal random variable with

zero mean and standard deviation σ in dB, that is, $X \sim \mathcal{N}(0, \sigma^2)$. All these parameters are determined for each environment and describe on average the distribution of RSSI at a point distant d from a transmitter. They are often obtained by collecting and processing RSSI measurements with linear regression techniques or maximum likelihood estimation (Roos et al., 2002). These techniques offer a minimum square error among the real data and the candidate log-distance model, which is expected to has the best fit.

For instance, suppose one has collected a considerable number of fingerprints from different places in the same environment. From the real data, a model is generated to represent the distribution of RSSI over the entire area. Often, for large environments, it is unlikely that only one model with some set of parameters precisely represents the signal variability in each arbitrary physical point. In this sense, although the aim of the model is to generalize the behavior of RSSI across the indoor space, some compartments might be possibly described by some other better models or representations. This notion is important, once the location prediction is directly related to the chosen model, and how this model is applied to some specific estimation problem.

Specifically in this work, the parameters of the log-distance path loss model are obtained by the use of the LinearRegression(x_c, y_c) function from the software Octave (Eaton et al., 2020), in which x_c is a function of the considered log-distances, and y_c the collected RSSI at location x_c. The function concerning the linear regression technique then returns the coefficients $PL(d_0)$ and α of the model, as well as the variance σ^2.

3.3 Bayesian Estimation

By knowing how to describe the RSSI variability from a log-distance model composed of a certain set of parameters, the Bayes theory helps us estimate the location of a target by knowing only the values in a set of fingerprints being collected in an undetermined position.

The model presented before can be slightly modified to describe the distribution

of RSSI at each point over the area:

$$r = P_t - PL(dB) = \{P_t - PL(d_0)\} - \left\{10\alpha \log\left(\frac{d}{d_0}\right) + X_\sigma\right\} \tag{3.2}$$

where r is the perceived power in the receiver device and P_t is the AP transmission power. It is important to notice that r is also a random variable, which can be represented by $r \sim \mathcal{N}(\mu_r, \sigma^2)$, in which μ_r is the expected value of the RSSI for a point in the environment:

$$\mu_r = P_t - PL(d_0) - 10\alpha \log\left(\frac{d}{d_0}\right) \tag{3.3}$$

The equations listed above describe the distribution of RSSI given a point distant d from an AP, which is known in the literature as the *likelihood function*, whose probability density function (p.d.f) is given by:

$$p(r|l) = \frac{1}{\sqrt{2\pi}\sigma} exp\left\{-\frac{1}{2}\left(\frac{r-\mu_r}{\sigma}\right)^2\right\} \tag{3.4}$$

where l is such that $d = \|l - l_{AP}\|$, with l and l_{AP} being the test point and the AP coordinates, respectively.

On the other hand, the main interest is to know the distribution of l given the RSSI information, which is obtained by the collected data. In this case, the *posterior function* contains the necessary information to estimate the location coordinate l. According to Bayes' rule:

$$p(l|r) = \frac{p(r|l)p(l)}{p(r)} \tag{3.5}$$

where $p(l)$ is the *prior* function and $p(r)$ is a normalizing factor given by the total

probability theorem:

$$p(r) = \int p(r|l')p(l')dl' \tag{3.6}$$

Equation (3.6) refers to the continuous case, in which l' represents each possible coordinate uniformly distributed over the area. Although it is computationally unfeasible to calculate this integral analytically, an approximation to the discrete form can be done (Honkavirta et al., 2009). That is, the area can be divided into many discrete coordinates as possible, treated here as the reference points (RPs). Likewise, the likelihood function is computed for each RP given. Thus, Equation (3.5) can be rewritten as:

$$p(l_i|r) = \frac{p(r|l_i)p(l_i)}{\sum_{j=1}^{m} p(r|l_j)p(l_j)} \tag{3.7}$$

where m is the number of RPs and $p(l_i|r)$ is the posterior function that relates the measure of RSSI r with location l_i, in which $i \in \{1, 2, ..., m\}$.

The equations we have seen so far take into account one RSSI sample from one AP only. However, $n > 1$ APs are considered in practical situations to improve accuracy, as it generates fewer ambiguities among the candidate RPs for the estimated location. In this case, the n-dimensional RSSI vector $\mathbf{r}$ is adopted instead of the one-dimensional r. Another strategy to improve accuracy is to collect a sufficient number of RSSI samples and take their mean for the estimation. According to the *strong law of large numbers*, the sample mean $\bar{\mathbf{r}}$ tends to its true value $\boldsymbol{\mu}_{\bar{\mathbf{r}}}$, as well as the *Tchebycheff's condition* states that the variance of the estimator of the mean $\bar{\sigma_n}^2$ tends to zero as $n \to \infty$ (Papoulis, 1991). Thereby, as variance diminishes, accuracy is improved due to fewer ambiguities in the estimation calculus.

Considering the multivariate Gaussian distribution (Tacq, 2010), the likelihood function already presented in Equation (3.4) can be rewritten as:

$$p(\mathbf{r}|l_i) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} exp\left\{-\frac{(\mathbf{r} - \boldsymbol{\mu}_{\mathbf{r}}^{(i)})^T \Sigma^{-1} (\mathbf{r} - \boldsymbol{\mu}_{\mathbf{r}}^{(i)})}{2}\right\} \tag{3.8}$$

where Σ is the covariance matrix, and $\boldsymbol{\mu}_{\mathbf{r}}^{(i)}$ the vector with expected values for the RSSI at location l_i. The exponential term of Equation (3.8) is known, when its root is taken, as the Mahalanobis distance. However, we consider the RSSI data provided by different APs as statistically independent, and a natural consequence is that the covariance matrix becomes diagonal. This way, the Mahalanobis distance reduces to the well-known Euclidean distance, becoming then:

$$p(\mathbf{r}|l_i) = \frac{1}{(2\pi\sigma^2)^{\frac{n}{2}}} exp\left\{ -\frac{\sum_{k=1}^{n}(r_k - \mu_{r_k}^{(i)})^2}{2\sigma^2} \right\} \tag{3.9}$$

Equation (3.9) represents the likelihood function of a vector containing n elements that correspond to the RSSI from each of the n APs. Next, the posterior function of Equation (3.7) can be finally presented in its vectorial shape:

$$p(l_i|\mathbf{r}) = \frac{p(\mathbf{r}|l_i)p(l_i)}{\sum_{j=1}^{m} p(\mathbf{r}|l_j)p(l_j)} \tag{3.10}$$

By knowing how to compute the probabilities for each RP, the final step is to find an estimator $\hat{\ell}$ for the position ℓ. One classic estimation procedure is to find the *maximum a posteriori estimate* $\hat{\ell}_{MAP}$, which simply gives the RP coordinate l_i that maximizes $p(l_i|\mathbf{r})$ in Equation (3.10):

$$\hat{\ell}_{MAP} = \underset{l_i}{\operatorname{argmax}}\, p(l_i|\mathbf{r}) \tag{3.11}$$

In other words, one seeks for the RP coordinate l_i in regards to which the sum presented into the exponential term in Equation (3.9) is minimized. This estimation is usually easy to determine (de Coulon, 1986) as well as it needs less computational effort. Other estimation variations are possible, as the weighted-mean, which simply is the normalized weighted sum of the probable locations.

3.4 K-means Clustering

Once we know how to represent the RSSI over an environment using the log-distance model and how to apply Bayes theory to estimate the location of an unknown set of fingerprints, the clustering step aims to group similar sets of fingerprints in order to assign for each set a more representative log-distance path loss model. Intuitively, similar fingerprints have the property of varying less than the average of all of them. In this case, a particular model is more likely to fit better to that specific set of RSSI vectors.

K-means clustering is one of the most traditional algorithms to gather similar instances (training examples) to a certain group. The parameter K determines the number of groups (or clusters) in which the data are separated. The partition is accomplished according to the euclidean distance among each instance and the existing centroids. Naturally, a centroid initialization is mandatory.

Next, an example of the algorithm is depicted to show how to split m instances into K different clusters. In general, the number of iterations R necessary for good convergence results is at least 100.

Algorithm 1: K-means Clustering

Data: number of clusters K, training set $\{x^{(1)}, x^{(2)}, ..., x^{(m)}\}$, and number of rounds R
Result: centroids μ and clusters indexes c assigned to each instance of training

```
φ ← 1
while φ < R do
    i ← 1
    while i < m do
        c^(i) ← index (from 1 to K) of cluster centroid closest to x^(i)
        i ← i + 1
    end
    j ← 1
    while j < K do
        μ_j ← average of points assigned to cluster j
        j ← j + 1
    end
    φ ← φ + 1
end
```

The K-means clustering algorithm applied to the proposed work aims to divide the training dataset into K different sets of fingerprints. Each set, naturally, has similar

fingerprints, i.e., RSSI vectors that are close to each other in terms of the Euclidean distance in the signal space. Each group is then assigned a specific log-distance model, which is obtained by employing the linear regression technique of the corresponding RSSI data.

3.5 Chapter Summary

This chapter presented the key theoretical aspects of the tools deployed in the proposed work. The main features of the fingerprint-based systems were addressed, such as the need for an offline phase to collect the RSSI signature of the indoor environment, and important concepts as training points and fingerprints. Also, the kNN was briefly described as one of the most deployed instance-based algorithms in such systems. Next, the log-distance path loss model was introduced as an alternative to describe the RSSI variability by taking into account its stochastic component. The Gaussian essence of this random component enables the RSSI to be associated with a probability density function, which, in turn, is a key concept for the theory of estimation using Bayes inference. Finally, the K-means clustering algorithm was described as the agent responsible for grouping similar instances of RSSI into clusters with associated log-distance path loss models.

4 Proposed Method

In this chapter, the proposed system is described. The approach closely follows the well-known fingerprint-based schemes, which are composed of an offline and an online phase. Each component from each phase is briefly discussed and the corresponding architecture depicted at first, and more technical details are presented further.

4.1 System Architecture

The proposed system consists of an offline phase and an online phase, as depicted in Fig. 2. In the offline phase, the log-distance parameters are determined and stored immediately after the clustering step. In the online phase, the model parameters corresponding to the most similar computed centroid are selected and used as a reference for the Bayesian estimation.

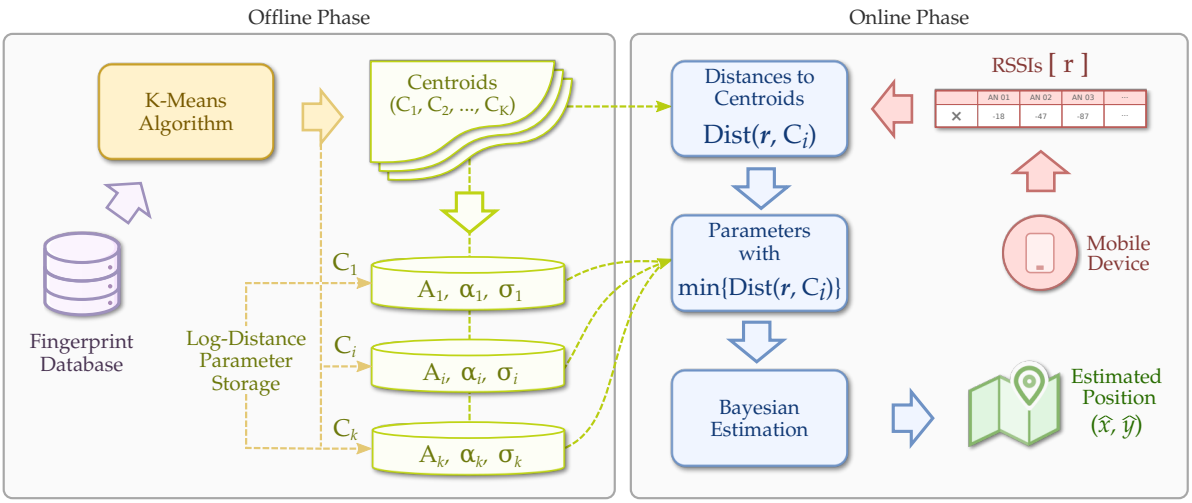

Figure 2 – Positioning system architecture composed of an offline phase and an online phase. Dist(r, C_i) refers to the euclidean distance between RSSI r and centroid C_i.

The figure above can be described as follows:

1. The offline phase, in summary, aims to train the model used in the presented work. It is composed of a fingerprint database, which contains all the fingerprints and their corresponding locations from the site-survey process. The K-means algorithm is then applied to the dataset in order to divide the fingerprints into K different groups. Each group has a corresponding centroid, which is simply a vector containing the average of the RSSI vectors assigned to the respective group. Also, the groups are represented by an specific log-distance model, which is obtained by linear regression of the corresponding RSSI data.

2. The online phase aims to deliver a reasonable location estimation of a mobile device in the environment. With a set of fingerprints whose location is still unknown, a centroid from the offline phase is assigned to the respective RSSI vector according to the similarity between them. The closest centroid, in this sense, is more likely to represent that particular set of fingerprints, and so it is the corresponding log-distance model. Then, with the candidate model and its parameters, it is possible to apply the Bayesian estimation to predict the final location.

Next, more details of each phase are presented.

4.2 Offline Phase

4.2.1 Fingerprint database

For every training point (TP) of the scenario, RSSI samples from each AP are collected. A fingerprint, in this case, is a composition of one RSSI sample from each AP in the environment. In other words, a fingerprint can be represented as a n-dimensional vector, in which n is the number of APs. Considering the number of fingerprints per training point as f, and the number of training points as m, the total number of collected fingerprints should be $n_{fp} = m \times f$. Here, besides considering the RSSI from the APs as features, the relative distance among the corresponding training points and the APs is

also included. In this sense, each fingerprint is now represented as a $2n$-dimensional vector.

4.2.2 Clustering

The algorithm used in this step is the K-means, which gathers the most similar fingerprints into K clusters. More specifically, due to its improvement on speed and accuracy, the K-means++ is deployed (Arthur and Vassilvitskii, 2007).

Most of the RSSI-based clustering solutions focus on grouping RSSI samples into different sets with similar elements each. Nevertheless, a cluster might contain samples from entirely different locations due to the high signal level variability over the indoor environment. In this sense, the addition of a fingerprint location constraint would somewhat help the clustering algorithm to group RSSI samples that are close to each other. Specifically, the logarithm of the distance among the APs and the fingerprints is added as a means of diminishing the probability that two fingerprints with significantly different relative positions to the APs belong to the same cluster. This attribute choice comes directly from the structure verified in the log-distance path loss model. Concretely, the K-means clustering algorithm is fed with the following matrix, whose rows and columns are typically known as instances and attributes, respectively:

$$F = \begin{pmatrix} r_{1,1} & \cdots & r_{1,n} & s_{1,1} & \cdots & s_{1,n} \\ r_{2,1} & \cdots & r_{2,n} & s_{2,1} & \cdots & s_{2,n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ r_{n_{fp},1} & \cdots & r_{n_{fp},n} & s_{n_{fp},1} & \cdots & s_{n_{fp},n} \end{pmatrix}, \tag{4.1}$$

where $r_{i,j}$ is the RSSI concerning the i-th fingerprint from the j-th AP, and $s_{i,j}$ refers to $\log d_{i,j}$, with $d_{i,j}$ as the distance between i-th fingerprint and j-th AP locations.

In the offline phase, the instances of F in Equation (4.1) are clustered according to some defined K. In the online phase, the selection of the corresponding cluster is accomplished by considering only the left half of F, since the only comparison object is the current RSSI vector. Later, one can verify that the addition of the position information

improves the accuracy of the system.

4.2.3 Parameters storage

After the preceding grouping process, the log-distance parameters for each cluster are stored by performing a simple linear regression of the received data according to the model described in Equation (3.2). In other words, each cluster which contains a certain set of fingerprints is associated with a specific log-distance path loss model that should represent the signal propagation over there. In practice, for each cluster centroid C_i, where $i \in \{1, 2, ..., K\}$, there is an associated set of parameters A_i, α_i, and σ_i, where $A_i = P_t - PL(d_0)_i$.

4.3 Online Phase

4.3.1 Cluster selection

The selection is performed by taking the minimum of the Euclidean distance between the current RSSI vector and each centroid stored previously:

$$c = \underset{i}{\operatorname{argmin}} \left\{ \sqrt{\sum_{j=1}^{n} (r_j - C_i)^2} \right\}, \tag{4.2}$$

where n is the number of APs, and $i \in \{1, 2, ..., K\}$. The log-distance parameters assigned to cluster c are then transfered to the model in Equation (3.2) for further processing.

4.3.2 Position estimation

With the log-distance path loss model assigned to the current RSSI vector **r**, it is possible to compute the probability associated with the chosen reference point l_i as

$$p(\mathbf{r}|l_i) = \frac{1}{(2\pi\sigma^2)^{\frac{n}{2}}} exp\left\{ -\frac{\sum_{k=1}^{n}(r_k - \mu_{r_k}^{(i)})^2}{2\sigma^2} \right\}, \tag{4.3}$$

where $i \in \{1, 2, ..., n_{rp}\}$, with n_{rp} as the number of RPs, and $\mu_{r_k}^{(i)}$ is the expected RSSI value at the i-th RP from the k-th AP.

By applying a variation of Equation (3.7), one possible estimation to the true location ℓ can be expressed as:

$$\hat{\ell} = \frac{\sum_{j=1}^{\beta} l_j p(\mathbf{r}|l_j)}{\sum_{j=1}^{\beta} p(\mathbf{r}|l_j)}, \tag{4.4}$$

where l_j is the $j-$th most probable location, whose p.d.f. is represented by $p(\mathbf{r}|l_j)$, computed by Equation (4.3). In this case, β is the number of reference points with location l_j to consider in the weighted estimation. Specifically, $\beta = 5$ is set in the validation experiments.

4.4 Chapter Summary

In this chapter, the main components of the proposed system were described in more detail. The offline phase is responsible for the model training, i.e., the division of the fingerprint database into different clusters represented by an unique log-distance path loss model each. The online phase, in turn, requests from the offline phase the model which is more likely to represent the current RSSI measurement for the estimation. In other words, given a certain dataset, composed of RSSI vectors, the basic location estimation process verified throughout this chapter is summarized as follows:

1. Grouping of similar fingerprints using the K-means clustering algorithm;

2. Mapping of each group to a representative log-distance path loss model;

3. Selection of the most similar group to represent a given unknown target with only RSSI information; and

4. Association with the corresponding log-distance model of the selected group for feeding the Bayesian estimation.

5 Results

In this chapter, the results of the proposed system are described and discussed. Firstly, the experimental testbed is presented. Next, the clustering target is slightly modified to improve the accuracy. It is also shown how the positioning error varies with the number of clusters. The corresponding evaluation metric, in this case, is the root mean square error (RMSE) in meters (m). Finally, a comparative performance analysis among the traditional Bayesian algorithm, the kNN, and the proposed KLIP solution is provided in terms of accuracy and processing time.

5.1 Experimental Testbed

To evaluate the performance of the solution, both online and offline phases were executed in a real-world experimental testbed. Fig. 3 shows the floorplan of the testbed: a 45 m $\times$ 16 m area with 11 rooms and 3 halls. The deployed infrastructure is based on BLE technology, from the transmitter devices (access points) to the receiver devices (wearables). The APs have a transmission power P_t of 0 dBm each and are spread over the site to provide good signal coverage. A total of 148 points, distant 2 m from each other and uniformly distributed throughout the floor, is selected to collect RSSI samples from the 14 available APs. The term 'point' refers to a real geographic point located in the floor plan. At each point, 50 samples are collected using 5 different receiver devices (10 samples each). The samples of each collection point are also split into 30 for training and 20 for tests.

It is important to highlight that the proposed solution does not necessarily use all 148 points to train the model. This scenario is specially useful to analyze the effect

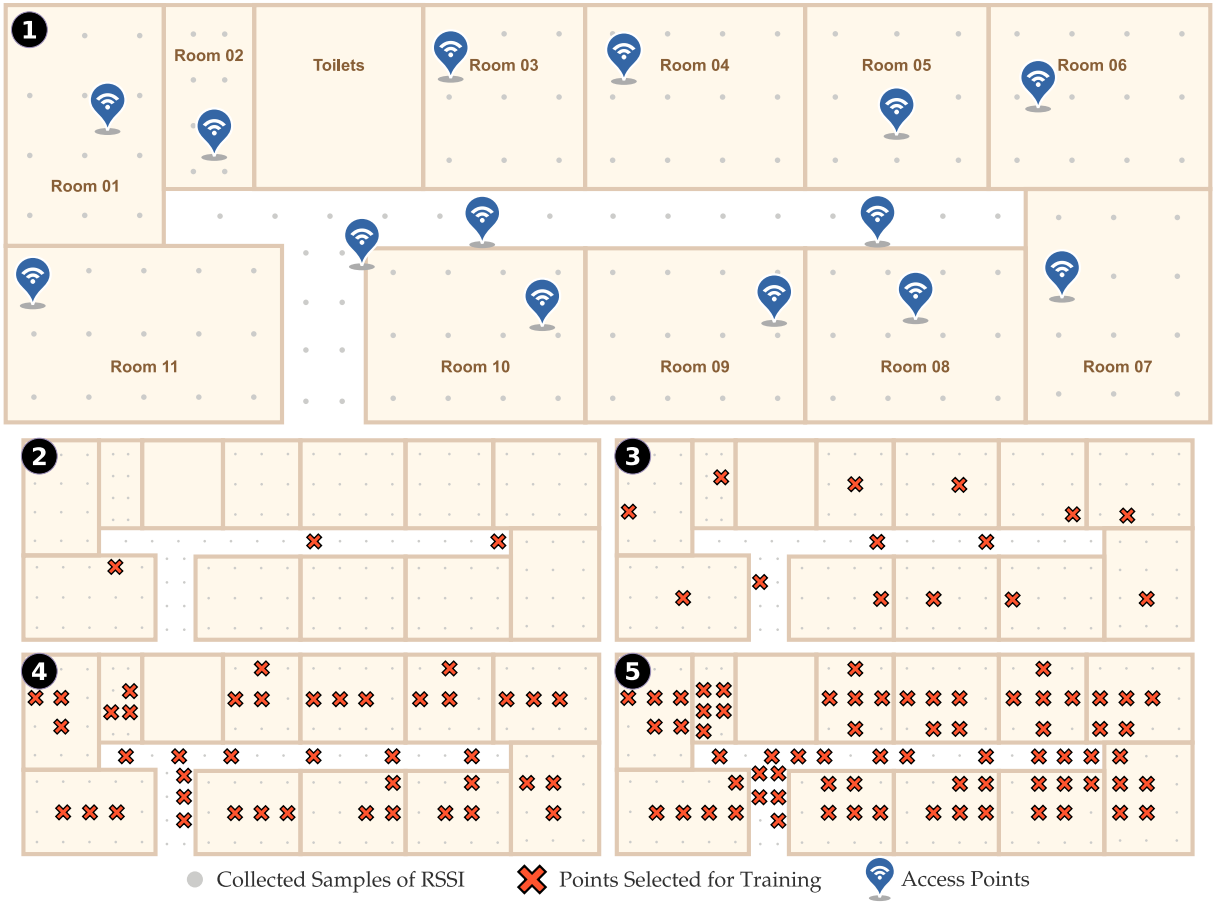

Figure 3 – Experimental testbed in different training scenarios. The upper half (1) shows 148 points (gray dots) where all the RSSI samples are collected. It also corresponds to the points selected for the test set, and the full training scenario. The lower half (2–5) represents four more distinct training scenarios, where each training point (TP) is depicted by the cross marks in red.

of the number of training points on the accuracy and the processing time for different positioning algorithms. The experiments are run over a 2-D problem, though the vertical distance among the APs on the ceiling and the RSSI collection points is considered as 2.5 m in the estimations. The upper half of Fig. 3 depicts the full training scenario, where all the collection points are used for training the model. The lower half of Fig. 3 depicts four more different training scenarios with 3, 14, 42, and 70 selected points for training, respectively.

The test set is composed of 296 RSSI vectors related to the 148 test points, as a result of using two different receiver devices. For each position estimation, the average of the 10 collected samples is taken to improve the system's accuracy.

In Table 2, a summary containing the essential information is provided regarding the experimental testbed. All data processing, system modeling, and tests are performed

using the Octave software packages (Eaton et al., 2020) on a Sony Vaio laptop (Windows 10, 64-bit operating system, 2.70 GHz Intel i7-7500U Processor and 8 GB RAM).

Table 2 – Summary of information about the experimental testbed

Parameter	**Value**
Indoor area	720 m^2
RSSI collection points	148
Training points (TPs)	{3,14,42,70,148}
Training devices	3
Test points	148
Test devices	2
Tests	296
Samples collected per device per point (training and test)	10

5.2 Clustering Attributes

Fig. 4 depicts the results for two approaches: using only one attribute $(r_{i,j})$, which is called the KLIP-R algorithm; and using two attributes $(r_{i,j}, s_{i,j})$, which is called the KLIP-RD algorithm. For both, the experiments are run by varying the number of training points from 3 to 148. The number of clusters considered for each run is described in Table 3, which corresponds to the minimum found positioning error.

Indeed, the addition of the distance attribute improves the positioning accuracy. For 3 and 14 training points, there is no difference between the KLIP-R and the KLIP-RD. However, from 42 training points onwards, there is a significant difference of up to 3.5% between the algorithms. As the number of training points increases, the distance information plays a more relevant role in distinguishing the clusters accordingly since the close similarity among RSSI samples becomes more frequent. For all the remaining analysis, the KLIP-RD is used to represent the KLIP itself.

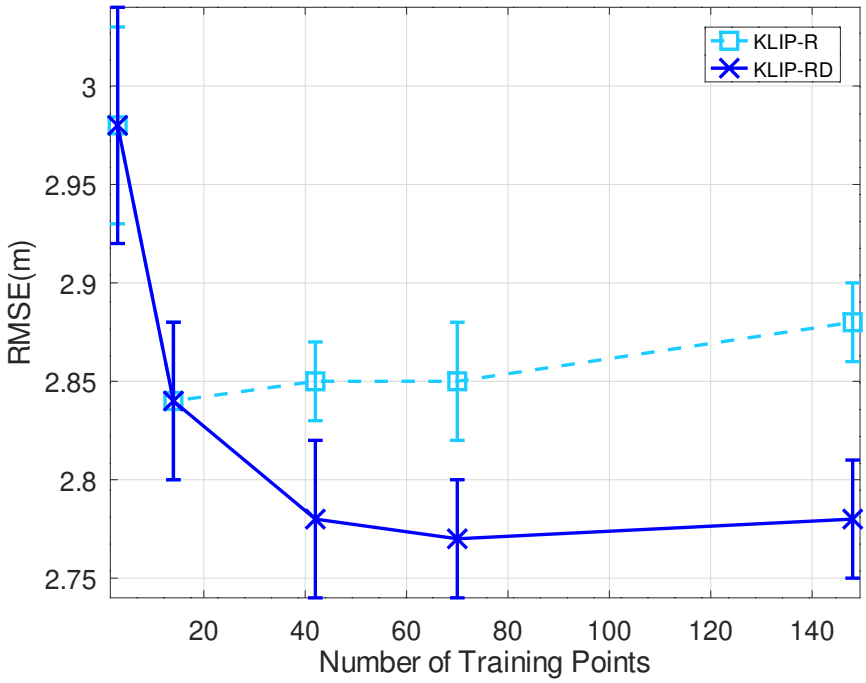

Figure 4 – Variation of the root mean square error (RMSE) with the size of the training dataset for different features used for clustering: RSSI-only (KLIP-R), and RSSI with distance attribute (KLIP-RD).

5.3 Number of Clusters

The number of clusters is a relevant parameter for reasonably good positioning accuracy. Experiments were performed for all sets of training points considered in the evaluation testbed. As an example to illustrate the impact of the number of clusters on the RMSE, one training point per room was chosen, which is equivalent to a total of 14 over the floor. The results are shown in Fig. 5. Visually, one can verify that there is no significant reduction in the error from 20 clusters onwards. For this particular scenario, any number of clusters close to 20 seems to be a reasonable choice. Also, compared to the traditional Bayesian algorithm, one can verify an improvement of about 8.7%. As for the other scenarios with different training dataset sizes, the accuracy behavior is very similar, and the optimal number of clusters does not exceed 36, as depicted in Table 3.

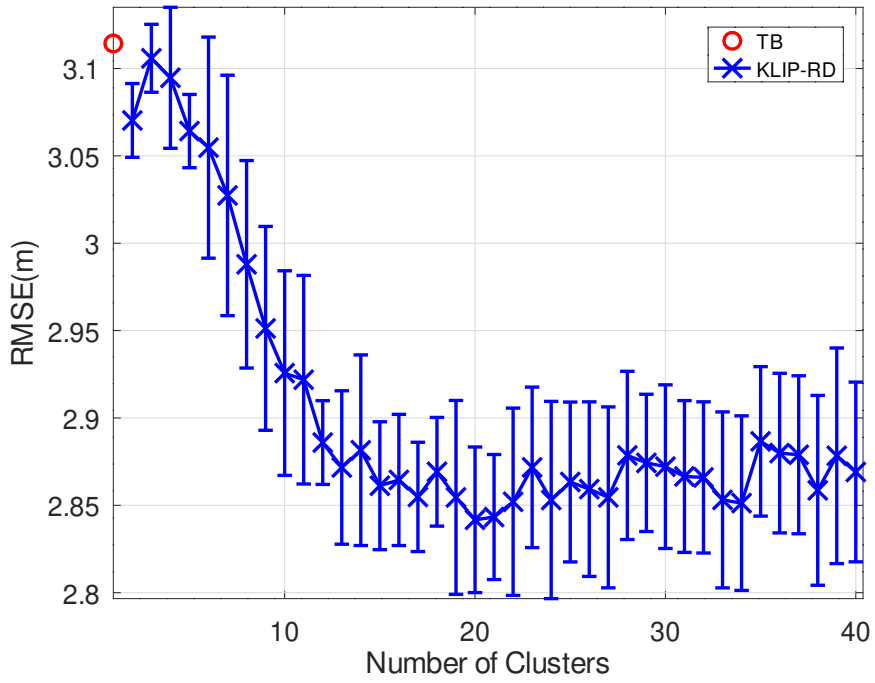

Figure 5 – Variation of the RMSE with the clusters computed by the K-means algorithm. The Traditional Bayesian (TB) algorithm is represented by the circle in red (1 cluster). The proposed KLIP is represented by the crosses in blue.

5.4 Comparative Performance Analysis

In this section, a comparative analysis is performed among three different positioning algorithms concerning the RMSE and the online processing time: the proposed KLIP (variant KLIP-RD), the traditional Bayesian (TB), and the kNN.

Table 3 – Performance Analysis: RMSE

#TPs	**RMSE** (m)			
	k**NN** (k)[1]	**TB**	**KLIP-R** (K)[2]	**KLIP-RD** (K)[2]
3	6.10 (35)	3.37	2.98±0.05 (29)	2.98±0.06 (25)
14	3.60 (7)	3.11	2.84±0.04 (27)	2.84±0.04 (21)
42	2.80 (37)	3.08	2.85±0.02 (21)	2.78±0.04 (25)
70	2.61 (49)	3.08	2.85±0.03 (26)	2.77±0.03 (36)
148	2.54 (36)	3.11	2.88±0.02 (24)	2.78±0.03 (34)

[1] The number of nearest neighbors k is indicated in parentheses for each run of the kNN algorithm.
[2] The number of clusters K is indicated in parentheses for each run of the KLIP algorithm.

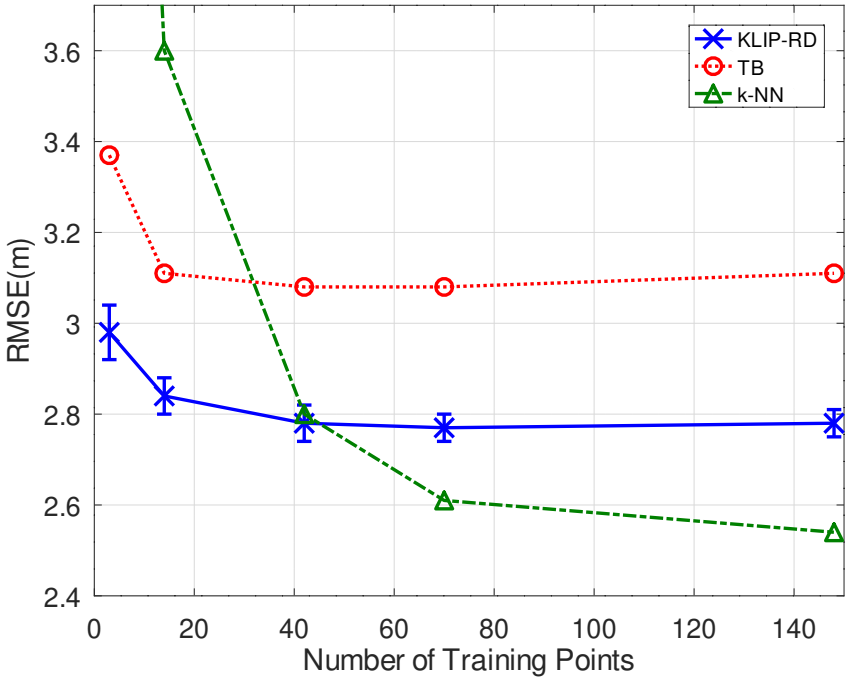

Figure 6 – Variation of the RMSE with the size of the training dataset for different positioning algorithms: the proposed KLIP, the traditional Bayesian (TB), and the classic kNN.

5.4.1 Positioning accuracy

Fig. 6 depicts the RMSE results for 3, 14, 42, 70, and 148 (full) training points using different positioning algorithms. For each training set, the results are plotted by considering the best accuracy performance for each algorithm. One can notice that KLIP outperforms TB for all the scenarios, whereas the performance over kNN is better for three scenarios, but only with reduced training dataset size. The results show an improvement of about 12% when comparing KLIP with TB for 3 training points. From 14 training points onwards, the improvement is, on average, 10%. When compared to kNN, KLIP is significantly more accurate for very small sets of training points, as for 3 and 14 ones. On the other hand, 70 training points are sufficient for the kNN to deliver better results. This is under what one should expect, as the increase in dataset size also increases the search space for the kNN to estimate more accurate results. Conversely, the increase in dataset size does not have the same impact on the KLIP performance, as the model regressor saturates with relatively few training points. In Table 3, the RMSE of each positioning technique is depicted for each scenario. For the kNN, specifically, one

can notice an apparent discrepancy of the obtained value of k for the scenario with 14 TPs. Actually, these optimal k's are not stable, and there is no pattern to be observed, as it depends on the set of considered fingerprints. Nevertheless, the observed differences in accuracy for different values of the parameter k at the top of the accuracy rank were small. Also, these differences did not affect the processing time of the implemented algorithm significantly.

5.4.2 Processing time

The computational cost is also a fundamental metric to measure the performance of IPSs when energy consumption and real-time applications are a stringent constraint. Fig. 7 shows the behavior of the required number of comparisons for each algorithm at the online (or estimation) phase.

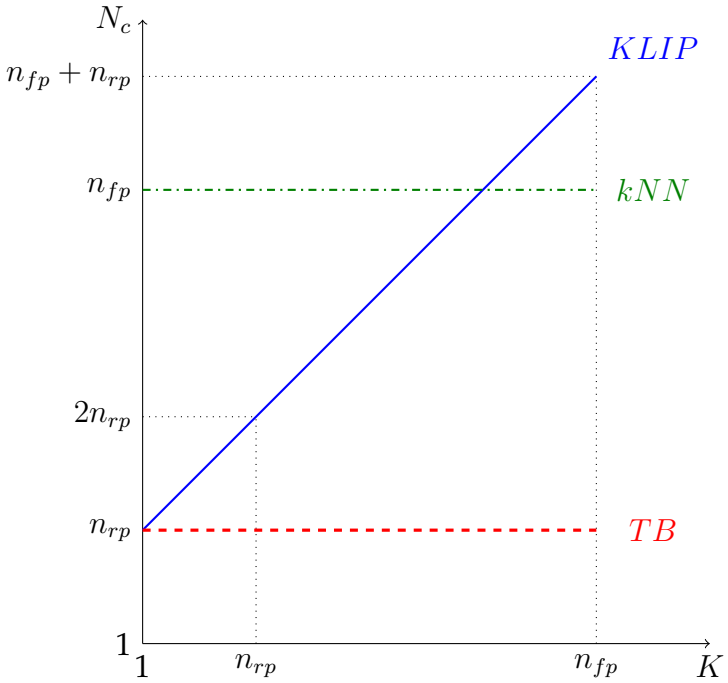

Figure 7 – Computational load comparison for each algorithm. N_c is the number of vector comparisons; K, the number of clusters; n_{rp}, the number of reference points; and n_{fp}, the number of fingerprints (total of samples).

A scalable scenario is described, where the number of fingerprints n_{fp} is much higher than the number of reference points n_{rp} used by the Bayesian estimation process, i.e., $n_{fp} >> n_{rp}$. Regarding the classic Bayesian algorithm, the number of comparisons is proportional to n_{rp}, which is the number of required operations to compute the probability for each reference point. Before the n_{rp} operations, the proposed KLIP performs a search among the K clusters assigned at the offline phase, which, in the worst case, can be set as the number of fingerprints n_{fp}. The kNN algorithm, in turn, performs a search among the n_{fp} samples to estimate the position with the smallest signal Euclidean distance. Most of the time, one should expect to deploy the KLIP algorithm with $K \leq n_{rp}$, which gives it a complexity $O(n_{rp})$. In the worst case, though, KLIP has a complexity $O(n_{fp}+n_{rp})$. In the experiments, $K < n_{rp}$, and the KLIP algorithm performs similarly to the classic Bayesian independently on the size of the training dataset. On the other hand, as the number of training points increases, the performance of the kNN decreases. The scenario is better depicted in Fig. 8, in which the average processing time per positioning estimation for each algorithm was measured through fifty runs. Table 4 provides the obtained measurements. In effect, the addition of the intermediate step for the KLIP does not require significantly more time when compared to the TB, as the number of clusters K is only a reduced fraction of the number of reference points n_{rp}. On the other hand, the kNN has the worst time performance as the number of training points increases, that is, as the number of fingerprints n_{fp} becomes the dominant time complexity factor.

Table 4 – Performance Analysis: Processing Time

#TPs	**Processing Time** (ms)		
	k**NN** (k)[1]	**TB**	**KLIP** (K)[2]
3	2.70±0.10 (35)	7.80±0.10	8.45±0.07 (25)
14	9.46±0.07 (7)	7.77±0.14	8.38±0.10 (21)
42	27.64±0.17 (37)	7.80±0.10	8.48±0.10 (25)
70	46.94±0.37 (49)	7.80±0.10	8.68±0.07 (36)
148	100.44±0.44 (36)	7.84±0.14	8.68±0.20 (34)

[1] The number of nearest neighbors k is indicated in parentheses for each run of the kNN algorithm.
[2] The number of clusters K is indicated in parentheses for each run of the KLIP algorithm.

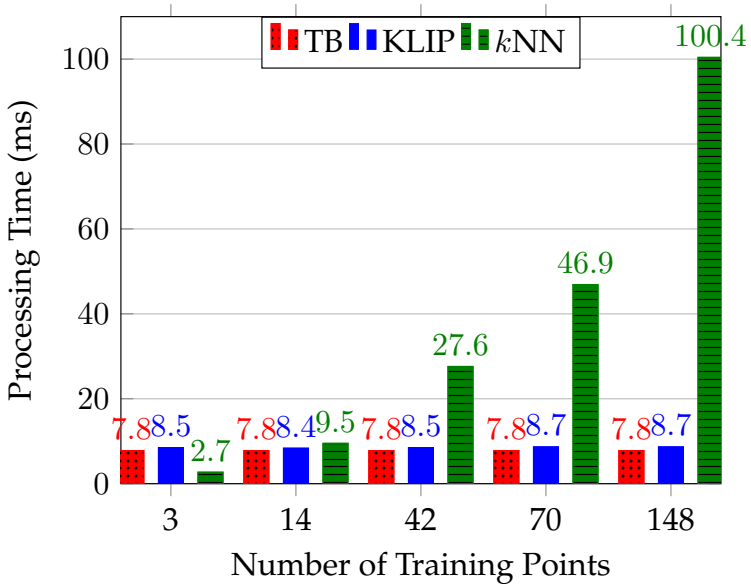

Figure 8 – Comparative analysis among the positioning algorithms in terms of average processing time.

5.5 Final Insights

In summary, one can infer that clustering the RSSI samples in different sets of log-distance models reduces the positioning error without requiring a significant computational load at the online phase. This is in accordance with the verified better performance of KLIP over TB. Moreover, the proposed method proves to be an alternative for reduced training size, as it outperforms the kNN in positioning accuracy. Although this does not hold for increasing training dataset size, the algorithm processing time is significantly shorter for KLIP over kNN. Specifically, the choice of 14 TPs, which is equivalent to one TP per floor compartment, is sufficient for the KLIP algorithm to provide a significant reduction in the training effort and in the positioning error, while keeping the processing time at a minimal level, when compared to both TB and kNN performances.

6 Conclusions

In this work, KLIP was presented, an indoor positioning solution that combines Bayesian inference with K-means clustering. The results verified throughout the work indicate that the high RSSI variability over the indoor environment can be better represented for more accurate position estimations without intensive site-survey by assigning subsets of RSSI samples to different sets of log-distance path loss models.

Although the work is not considered a classic fingerprint-based method, a fingerprint database was built to train the log-distance models in different training scenarios. From each group generated by the implementation of the K-means clustering algorithm, one model was associated with the corresponding cluster, and the model parameters were estimated by means of linear regression techniques. This step constitutes what was called the offline phase here. The estimation phase, on the other hand, was achieved by the implementation of two steps: the cluster selection and the location estimation itself. The first associated the RSSI measurement to a specific cluster represented by its centroid and its corresponding model in the offline phase. The latter used the assigned log-distance model as the basis for the Bayesian estimation algorithm.

All experiments were conducted in a real-world testbed and over a Bluetooth Low Energy (BLE)-based infrastructure. The indoor space comprises an area of approximately 720 m^2 with eleven rooms and three halls. The BLE access points were spread over the environment to cover the majority of the considered area. From 148 RSSI collection points uniformly distributed throughout the indoor site, five different training scenarios (3, 14, 42, 70, and 148 training points) were considered for evaluating the KLIP performance. Each scenario simulated a different training effort, and how the positioning error and the online processing time vary with this parameter. Compared to the traditional probabilistic approach based on Bayes theory, a positioning accuracy

improvement of up to 12% was verified in a minimum training effort scenario, and of 10%, on average, for the rest of the training scenarios. Also, when compared to the traditional kNN, the proposed system performed much more accurately for small training dataset sizes, although an inferior performance was observed in a full training scenario. In this sense, the increase in the search space for the kNN is beneficial, whereas it does not apply for the model-based KLIP, which has its accuracy early saturated with relatively few training points. Regarding the algorithm processing time during the estimation phase, the proposed system performed closely to the typical Bayesian approach, and approximately constant with the increase of training points, which shows robustness in this regard. In other words, KLIP is robust concerning the processing time, as this parameter is virtually invariant with the dataset size. This also indicates the KLIP system is scalable in terms of energy consumption, as less battery-life is needed for short processing time, and in terms of real-time applications, as the time required for the estimation process is sufficiently short for rapid location updates.

6.1 Limitations and Future Work

One of the main limitations of the work is the difficulty of knowing the minimum number of training points needed to achieve the maximum achievable accuracy. This is surely an important topic for the research area, and very challenging, due to the great diversity of indoor environments and their specificities. Similarly, the optimal number of clusters for each scenario was not addressed in this work, which is, by the way, an open research topic for scientists that work with the general clustering problem.

For future work, the research moves towards the exploitation and enhancement of the offline clustering step and the optimization of key parameters (as the number and the disposal of reference points) of the probability-based algorithm to reduce both the positioning error and the online computational load. A previous work published during the pursuing of this master's thesis (Pinto et al., 2021) addressed the impact of the number of reference points on accuracy when using the Bayesian estimation for the design of IPSs, and much of what was presented in the corresponding paper can serve

as the starting point for the continuity of the current research. In this sense, it is possible to build strategies for minimizing the number of reference points while keeping the positioning errors at minimal levels. Also, the geometry of the deployment of reference points over the environment should be an important variable to consider when dealing with room-level accuracy applications.

Bibliography

Alfakih, M., Keche, M., Benoudnine, H., and Meche, A. (2020). Improved gaussian mixture modeling for accurate wi-fi based indoor localization systems. *Physical Communication*, 43:101218. pages 8, 10

Alraih, S., Alhammadi, A., Shayea, I., and Al-Samman, A. M. (2017). Improving accuracy in indoor localization system using fingerprinting technique. In *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 274–277. pages 3

Altintas, B. and Serif, T. (2011). Improving rss-based indoor positioning algorithm via k-means clustering. In *17th European Wireless 2011 - Sustainable Wireless Technologies*, pages 1–5. pages 3

Arthur, D. and Vassilvitskii, S. (2007). K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, page 1027–1035, USA. Society for Industrial and Applied Mathematics. pages 21

Assayag, Y., Oliveira, H., Souto, E., Barreto, R., and Pazzi, R. (2020). Indoor positioning system using dynamic model estimation. *Sensors*, 20(24). pages 9, 10

Bahl, P. and Padmanabhan, V. N. (2000). Radar: an in-building rf-based user location and tracking system. In *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*, volume 2, pages 775–784 vol.2. pages 2

de Coulon, F. (1986). *Signal Theory and Processing*. Artech House, Dedham, MA, 1st. edition. pages 16

Eaton, J. W., Bateman, D., Hauberg, S., and Wehbring, R. (2020). *GNU Octave version 5.2.0 manual: a high-level interactive language for numerical computations*. pages 13, 27

Farid, Z., Nordin, R., and Ismail, M. (2013). Recent advances in wireless indoor localization techniques and system. *Journal Comp. Netw. and Communic.*, 2013:185138:1–185138:12. pages 1

Gu, Y., Lo, A., and Niemegeers, I. (2009). A survey of indoor positioning systems for wireless personal networks. *IEEE Communications Surveys Tutorials*, 11(1):13–32. pages 4, 2, 12

Han, K., Xing, H., Deng, Z., and Du, Y. (2018). A rssi/pdr-based probabilistic position selection algorithm with nlos identification for indoor localisation. *ISPRS International Journal of Geo-Information*, 7(6). pages 7, 10

Hoang, M. T., Yuen, B., Dong, X., Lu, T., Westendorp, R., and Reddy Tarimala, K. (2020). Semi-sequential probabilistic model for indoor localization enhancement. *IEEE Sensors Journal*, 20(11):6160–6169. pages 8, 10

Honkavirta, V., Perälä, T., Ali-Löytty, S., and Piché, R. (2009). A comparative survey of wlan location fingerprinting methods. *2009 6th Workshop on Positioning, Navigation and Communication*, pages 243–251. pages 15

Klus, L., Quezada-Gaibor, D., Torres-Sospedra, J., Lohan, E. S., Granell, C., and Nurmi, J. (2020). Rss fingerprinting dataset size reduction using feature-wise adaptive k-means clustering. In *2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pages 195–200. pages 3

Li, A., Fu, J., Shen, H., and Sun, S. (2021). A cluster-principal-component-analysis-based indoor positioning algorithm. *IEEE Internet of Things Journal*, 8(1):187–196. pages 3

Li, L., Guo, X., and Ansari, N. (2020). Smartloc: Smart wireless indoor localization empowered by machine learning. *IEEE Transactions on Industrial Electronics*, 67(8):6883–6893. pages 8, 10

Li, Y., Williams, S., Moran, B., and Kealy, A. (2019). A probabilistic indoor localization system for heterogeneous devices. *IEEE Sensors Journal*, 19(16):6822–6832. pages 8, 10

Liu, F., Liu, J., Yin, Y., Wang, W., Hu, D., Chen, P., and Niu, Q. (2020). Survey on wifi-based indoor positioning techniques. *IET Communications*, 14(9):1372–1383. pages 1, 3

Liu, H., Darabi, H., Banerjee, P., and Liu, J. (2007). Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*

(Applications and Reviews), 37(6):1067–1080. pages 1

Liu, W., Fu, X., and Deng, Z. (2016). Coordinate-based clustering method for indoor fingerprinting localization in dense cluttered environments. *Sensors*, 16(12). pages 3

Macagnano, D., Destino, G., and Abreu, G. (2014). Indoor positioning: A key enabling technology for iot applications. In *2014 IEEE World Forum on Internet of Things (WF-IoT)*, pages 117–118. pages 1

Man, D., Bing, L., and Lv, J. (2020). Indoor localization algorithm based on attribute-independent weighted naive bayesian. *Proceedings of the 2020 International Conference on Cyberspace Innovation of Advanced Technologies*. pages 3

Njima, W., Ahriz, I., Zayani, R., Terre, M., and Bouallegue, R. (2017). Smart probabilistic approach with rssi fingerprinting for indoor localization. In *2017 25th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pages 1–6. pages 7, 10

Papoulis, A. (1991). *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 3rd edition. pages 15

Pinto, B. H. O. U. V., de Oliveira, H. A. B. F., and Souto, E. J. P. (2021). Factor optimization for the design of indoor positioning systems using a probability-based algorithm. *Journal of Sensor and Actuator Networks*, 10(1). pages 35

Rappaport, T. S. (2002). *Wireless Communications: Principles and Practice*, volume 2. Prentice-Hall, Upper Saddle River, N.J, 2nd edition. pages 3, 12

Ren, J., Wang, Y., Niu, C., Song, W., and Huang, S. (2019). A novel clustering algorithm for wi-fi indoor positioning. *IEEE Access*, 7:122428–122434. pages 3

Roos, T., Myllymaki, P., and Tirri, H. (2002). A statistical modeling approach to location estimation. *IEEE Transactions on Mobile Computing*, 1(1):59–69. pages 13

Soares Lima, M. W., Fernandes de Oliveira, H. A. B., dos Santos, E. M., de Moura, E. S., Costa, R. K., and Levorato, M. (2018). Efficient and robust wifi indoor positioning using hierarchical navigable small world graphs. In *2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)*, pages 1–5. pages 1

Subedi, S. and Pyun, J.-Y. (2020). A survey of smartphone-based indoor positioning system using rf-based wireless technologies. *Sensors*, 20(24). pages 3

Tacq, J. (2010). Multivariate normal distribution. In Peterson, P., Baker, E., and McGaw, B., editors, *International Encyclopedia of Education (Third Edition)*, pages 332 – 338.

Elsevier, Oxford, third edition edition. pages 15

Torres-Sospedra, J., Quezada-Gaibor, D., Mendoza-Silva, G. M., Nurmi, J., Koucheryavy, Y., and Huerta, J. (2020a). New cluster selection and fine-grained search for k-means clustering and wi-fi fingerprinting. In *2020 International Conference on Localization and GNSS (ICL-GNSS)*, pages 1–6. pages 3

Torres-Sospedra, J., Richter, P., Moreira, A., Mendoza-Silva, G., Lohan, E., Trilles, S., Matey-Sanz, M., and Huerta, J. (2020b). A comprehensive and reproducible comparison of clustering and optimization rules in wi-fi fingerprinting. *IEEE Transactions on Mobile Computing*, pages 1–1. pages 3

Wu, T., Xia, H., Liu, S., and Qiao, Y. (2019). Probability-based indoor positioning algorithm using ibeacons. *Sensors*, 19(23):5226. pages 8, 10

Youssef, M. and Agrawala, A. (2005). The horus wlan location determination system. In *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*, MobiSys '05, page 205–218, New York, NY, USA. Association for Computing Machinery. pages 2

Zafari, F., Gkelias, A., and Leung, K. K. (2019). A survey of indoor localization systems and technologies. *IEEE Communications Surveys Tutorials*, 21(3):2568–2599. pages 3

Zhang, C., Qin, N., Xue, Y., and Yang, L. (2020). Received signal strength-based indoor localization using hierarchical classification. *Sensors*, 20(4). pages 3

Zhong, Y., Wu, F., Zhang, J., and Dong, B. (2016). Wifi indoor localization based on k-means. In *2016 International Conference on Audio, Language and Image Processing (ICALIP)*, pages 663–667. pages 3