Development of Localization Algorithms for a WiFi Based Indoor Positioning System with Machine Learning Techniques

Lu Xiaoxuan

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirements for the degree of
Master of Engineering

2015

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

....

Date Lu Xiaoxuan

Acknowledgements

I wish to express my heartiest gratitude to my supervisor, Prof. Xie Lihua, for his unconditional support, professional guidance and encouragement during my master study. What I appreciated most were the opportunities and the freedom he has offered to me for finding the research topics I am interested in. Without his great enthusiastic help, I would not be able to complete this thesis.

Thanks also to Professor Huang Guangbin for his collaborations and fruitful discussions. I am always impressed by his expertise on grasping decisive ideas for solving problems and his insights in his research areas.

I also would like to show my gratitude to the members of Sensor Network Lab and other friends who have helped and supported me during my study. Particularly, thanks go to my seniors, Mr. Zou Han and Dr. Jiang Hao, who helped me to understand quite a lot of critical problems.

Meanwhile, I would like to thank my parents, for their endless love, support and self-sacrifices. They always manage to bring me the best education opportunities than I deserve, this thesis for you and I promise this won't be my last thesis.

Abstract

In recent years, the prevalence of mobile devices and the popularity of social networks have spurred extensive demands on Location Based Services (LBSs). Whereas GPS has been extensively adopted in outdoor positioning, it can't provide accurate enough indoor positions due to non-line-of-sight (NLOS) transmission channels between the receiver and the satellite in indoor environments. Thus, developing an Indoor Positioning System (IPS) that is capable of providing reliable and accurate LBSs is widely studied. A large body of WiFi-fingerprinting based indoor localization solutions emerge as WiFi is accessible for users and cost-efficient for developers. On the other hand, it has been demonstrated in literature that machine learning techniques can be applied to IPSs yielding satisfactory localization results in real-time.

In this thesis, we aim to develop accurate and reliable localization algorithms for WiFi based IPS by machine learning (ML) techniques. We model the indoor positioning problem under a non-parametric stochastic framework, and modify the well-known ML tool, extreme learning machine (ELM), to achieve the above goal.

Firstly, under the assumption that noises merely lie in input data, we modify ELM by introducing a dead zone, which is called DZ-ELM, and integrate it into our IPS. We analyse the consistency of DZ-ELM for different types of disturbances. Experimental results show that the DZ-ELM based IPS can not only provide higher accuracy, but also improve the repeatability of IPSs.

Secondly, when assuming that noises lie in both input and output data, we exploit the fact that feature mapping in ELM is known to users to develop two kinds of

robust ELM (RELM) based on second order cone programming. The simulation and real-world indoor localization experimental results both demonstrate that the proposed algorithm can not only improve the accuracy and repeatability, but also reduce the deviations and worst case errors of IPSs compared with other baseline algorithms.

Lastly, beyond IPSs, we employ clustering algorithms to build up a clustering based zonal occupancy monitoring system by existing WiFi infrastructures to model the distribution of indoor occupants. The system is lightweight and free of calibration, which can be further incorporated in real applications such as optimization and control of building heating, ventilating, and air conditioning (HVAC) systems.

Symbols and Acronyms

Symbols

$\mathbb{C}$	set of complex numbers
$\mathbb{R}$	set of real numbers
$\mathbb{R}^+$	set of positive real numbers
$\mathbb{R}^N$	set of real N dimensional vectors
$\mathbb{R}^{M\times N}$	set of real $M \times N$ dimensional matrices
$\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{\Phi}$, $\cdots$	matrices
$\boldsymbol{x}$, $\boldsymbol{y}$, $\cdots$	vectors
x, y, $\cdots$	scalars
$\boldsymbol{A}_{i,:}$	ith row of $\boldsymbol{A}$
$\boldsymbol{A}_{:,j}$	jth column of $\boldsymbol{A}$
A_{ij}	ijth entry of $\boldsymbol{A}$
$\boldsymbol{A}^T$	transpose of $\boldsymbol{A}$
$\boldsymbol{A}^H$	complex transpose of $\boldsymbol{A}$
x_i	ith entry of $\boldsymbol{x}$
$\|\cdot\|_2$	spectral norm for a matrix, or Euclidean norm for a vector
$\|\cdot\|$	determinant for a matrix, or absolute value for a scalar, or cardinality for a set
$\operatorname{diag}(\cdot)$	a diagonal matrix if applied to a vector with its diagonal entries formed by the vector, or if applied to a matrix, a vector composed of diagonal entries of the matrix
$\operatorname{supp}\{\boldsymbol{x}\}$	support of $\boldsymbol{x}$ defined as $\{i : x_i \neq 0\}$

$\|\boldsymbol{x}\|_p$	ℓ_p norm ($p \geq 1$) or pseudo-norm ($0 < p < 1$) of $\boldsymbol{x}$ defined as $(\sum_i \|x_i\|^p)^{\frac{1}{p}}$
$\|\boldsymbol{x}\|_0$	ℓ_0 pseudo-norm of $\boldsymbol{x}$ defined as the number of nonzero entries
$\|\boldsymbol{x}\|_\infty$	ℓ_∞ norm of $\boldsymbol{x}$ defined as $\max\{\|x_i\|\}$
$\langle \boldsymbol{x},\ \boldsymbol{y} \rangle$	inner product of $\boldsymbol{x}$ and $\boldsymbol{y}$
$\boldsymbol{x} \succeq \boldsymbol{y}$	$x_i \geq y_i$ for all i
$E\{\cdot\}$	expectation
$Var\{\cdot\}$	variance

Acronyms

IPS	indoor positioning system
LBS	location based service
AP	access point
ML	machine leanring
MMSE	Minimum mean square error
HVAC	heating, ventilating, and air conditioning
SLFN	single-hidden layer feedforward neural network
ELM	extreme learning machine
RELM	robust extreme learning machine
OS-ELM	online sequential extreme learning machine
CTM	close to mean
SR	small residual
SVR	support vector regression
DZ	dead zone
SOCP	second-order cone programing
RSS	received signal strength
MRSE	mean root square error
WCE	worst case error

REP	repeatability
STD	stand deviation
FSFP	clustering by fast searching and finding of density peak

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Motivation and Objective

In recent years, the prevalence of mobile devices and the popularity of social networks have spurred massive demands on Location Based Services (LBSs). Whereas GPS has been extensively adopted in outdoor positioning, it can't provide accurate enough indoor positions due to NLOS transmission intermediate between the receiver and the satellite in indoor environments. Therefore, great efforts have been devoted to developing Indoor Positioning Systems (IPSs) so as to enable reliable and precise indoor positioning and navigation. As shown in Figure 1.1, IPS has been recognized as a crucial component in numerous applications such as asset tracking, logistics, security, tourism and security.

In the past two decades, various wireless communication technologies, such as Ultra-Wideband (UWB), Radio Frequency Identification (RFID), IEEE 802.11 (WiFi) and so on, have been exploited for indoor positioning and navigation services [3–5]. Due to the rapid development of miscellaneous wireless communication technologies to accommodate LBSs, indoor localization systems have been broadly developed and numerous algorithms have been studied recently [5, 6].

Unlike other wireless technologies, such as Ultra-Wideband (UWB) and Radio Frequency Identification (RFID), which require the deployment of extra infrastructures,

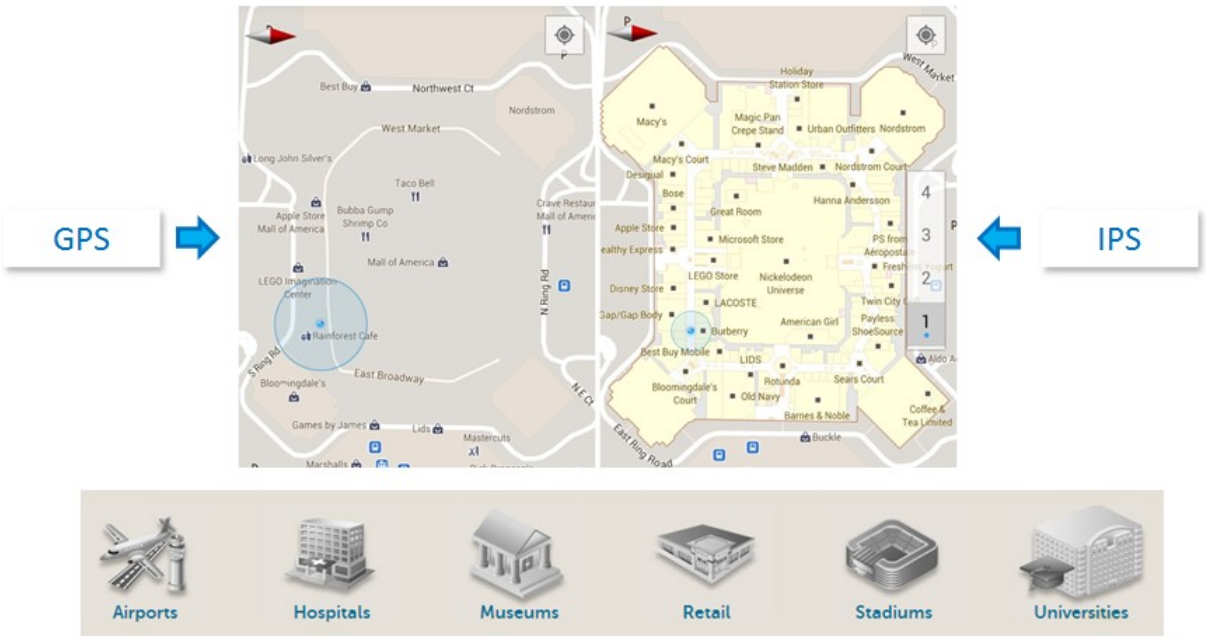

Figure 1.1: Indoor Positioning System [1]

the existing IEEE 802.11 network infrastructures, such as WiFi routers, are widely available in a large number of commercial and residential buildings. In addition, nearly each mobile device now is WiFi enabled [7].

On the other hand, reducing energy demands is a hot topic in the interdisciplinary research of location based services and smart buildings.

Building energy usage accounts for an increasing proportion of the total energy consumption. It's reported that the building electricity consumption has occupied 40% of the total electricity consumption in United States in 2005 and 33% in Singapore in 2010 respectively [8,9]. Currently, most HVAC systems assume a maximum occupancy status of the room during normal working periods, and a minimum occupancy status at night. This results in energy waste because the occupancy degree of rooms is conditioned to the number of occupants who are actually occupying the area [10]. The knowledge of occupancy levels in discrete zones within a building offers the potential of significant energy savings when coupled with zonal control of building services [11, 12].

The objective of this research is twofold: firstly, we aim to develop accurate and

reliable indoor positioning systems to locate and track devices and human being which can be useful in smart buildings. Portable devices such as mobile phones will be used to capture occupant presence and behaviors and to understand human activity patterns. Extreme learning machine (ELM), a cutting-edge machine learning technique, will be integrated into positioning algorithms to develop and test our localization systems that can provide high performance in terms of time and accuracy. Secondly, for the sake of energy saving, we also aim to develop a WiFi based occupancy monitoring system which can facilitate the optimization and control of HVAC systems.

1.2 Contributions

Our team developed a WiFi-based IPS. We have deployed this system in the Internet of Things Lab in NTU. Based on our experimental results, our system can provide around 2.2m localization accuracy consistently in different indoor scenarios.

Besides the development of IPS, ELM and its variants based indoor localization algorithms were proposed. In particular, the robustness of IPS is our primary concern. Due to ambient dynamics, signal variations (outliers) and manual observational errors, the performance of IPS will be affected. Thus, the aim of our work is to develop robust algorithms to overcome the effect of these uncertainties.

At the first hand, ELM with dead zone (DZ-ELM) is proposed to enhance the robustness against the ambient dynamics and outliers mixed into the training data set. Our simulation and experimental results both demonstrate that DZ-ELM can adapt the IPS to new environments and have better accuracy when facing different types of noises compared with the basic ELM based IPS.

Next, we incorporate all the adverse factors, e.g. manual observational errors, signal variations and so on, as uncertainties lying in input and output data, which motivates us to apply optimization techniques to address the robust localization problem. Two robust ELMs (RELMs) based on second order cone programming are

proposed under this framework. Simulation under different perturbations and real-world experiments both show that the proposed algorithms can not only improve the accuracy and repeatability, but also reduce the deviations and worst case errors of IPSs.

In the end, we employ a kind of density oriented clustering algorithm to firstly derive centroids from the crowds (clusters) and then classify them into corresponding zones. This algorithm can be suitable to indoor positioning systems in large-scale scenarios, where the area of the indoor plot and the volume of its occupants are extremely large. Experimental results show that our approach can accurately provide the indoor occupancy distribution.

1.3 Organization of the Thesis

Related works are reviewed in *Chapter 2*, the structure and overview of WiFi based IPS and preliminaries of ELM will be given in *Chapter 3*. *Chapters 4–6* present our main contributions. More specifically,

Chapter 4 presents the proposed extreme learning machine with dead zone (DZ-ELM) and its application to our indoor positioning system.

Chapter 5 is devoted to developing a robust extreme learning machine (RELM) and its integration with our indoor positioning system. Some of the results can be found in the paper published on [13].

Chapter 6 is devoted to developing the system for monitoring the distribution of indoor occupants via machine learning techniques.

Chapter 7 presents conclusions and highlights some potential future research directions.

Chapter 2

Literature Review

Accurate and inexpensive indoor localization is an indispensable component of location-based services. Intensive research has been done on various aspects of indoor positioning. Section 2.1 – Section 2.2 are devoted to a comprehensive review of recent advances in indoor localization, from the angles of sensing modalities and indoor positioning algorithms.

In the meanwhile, zonal occupancy distribution is informative for the optimization and control of HVAC systems. Section 2.3 presents the related work of zonal occupancy monitoring systems.

2.1 Sensing Modalities

2.1.1 Infrared (IR)

The range of infrared (IR) wavelength is from 700 nm to 1 mm, and the corresponding frequency range is from 430 THz to 300 GHz. Its wavelengths are longer than visible light, which makes IR indoor positioning systems less intrusive than visual-based indoor positioning systems. There are basically three methods of using infrared signals for indoor localization: active infrared beacons, infrared imaging using natural radiation and artificial light sources.

There are various methods to track or detect human beings by using infrared technology in the indoor environments. Artificial IR light source system can provide sub-mm localization accuracy. System based on active beacons and natural radiation are employed for providing coarse location estimation or computing the occupancy distribution of indoor environments.

The high deployment cost of extra infrared infrastructure is the main problem of lR in commercial applications.

2.1.2 Cameras

Cameras are able to provide high resolution of space, and accommodate information regarding objects in a scene, including texture, shape, color, size, and so on. The fast development of computer vision technology makes cameras widely adopted by robots for simultaneous localization and mapping (SLAM) tasks. In the meanwhile, high-dimensionality of images and video streams also leaves them much more difficult to analyse than signals from a majority of other technologies [14].

However, camera based modalities face the privacy problem and suffer from environmental conditions such as lighting and presence of occluding objects.

2.1.3 Inertial measurement unit (IMU)

The IMU, is a device that is capable of measuring an entity's velocity, gravitational forces and orientation of moving by gyroscopes (angular velocity sensors) and accelerometers (acceleration sensors). Due to the poor agility of gyroscopes in mobile devices, some devices also incorporate magnetic field sensors as a compass (magnetometers) in their IMU function. Information provided from the IMU's sensors allows one to track an entity's position, via methods such as dead reckoning (DR).

Although it's reported to be more accurate for IMU if it's foot-mounted with indoor occupants, it's not practical and economic. A large body of indoor localization

researchers leverage the embedded IMUs in mobile devices, whose accuracy falls dramatically as different placement of devices by holders can result in different measurements. Accumulated errors are often reported in cases that merely the IMUs in devices are utilized for indoor positioning.

2.1.4 Bluetooth low energy (BTLE)

BTLE is an emerging low power ($1mW$ typical transmitter output) technology that operates in the $2.4GHz$ industrial, scientific and medical (ISM) band. Compared with classic Bluetooth 2.2, BTLE is capable of interfacing with low-power sensors, e.g., smart watches, without long handshake period. Gaussian frequency-shift keying (GFSK) is adopted in it to make transmit and decode much simpler. From the perspective of indoor positioning, the superiority of BTLE lies in its frequency hopping mechanism and high sampling rate, which can alleviate the interference and multipath effect. Almost all modern smart phones, (Apple 6, Nexus 5 and Samsung S3) are equipped with BTLE transceivers, which justify its commercial value.

BTLE enabled devices are very low power - a device sending beacons at a rate of 10 Hz can run for over a year off a coin cell. Compared with other radio based technologies, they are substantially cheaper. Nevertheless, it cannot provide users with Internet services as WiFi, which is the main challenge BLTE faces.

2.1.5 Ultra-Wideband (UWB)

Ultra-Wideband (UWB) is a radio technology for high-bandwidth short-range communication with strong multipath resistance and penetrability through different building materials, which is beneficial for indoor ranging, localization and tracking. Different with narrowband operation, UWB waves occupy a large frequency bandwidth ($>$ 500 MHz). Its harmfulness to the human body is alleviated by the low power spectral density, which also facilitate restricting the interference of UWB signals with other narrowband receivers [2].

One major advantage of using UWB is that it is capable of penetrating construction materials, e.g., wood, concrete and glass. This is favorable for indoor localization as it capacitate ranging under non-line-of-sight transmission and enables inter-room distance measurements. Large bandwidth translates into a high resolution in time and consequently in range is another superiority of using UWB for ranging [2]. Favored by the short duration (less than 1ns) of UWB pulses, the reflected signals from the original signals can be filtered, which largely improves its accuracy.

2.1.6 Radio Frequency Identification (RFID)

RFID is a method of retrieving and storing data through electromagnetic transmission to a RF compatible integrated circuit. RFID-based indoor positioning systems usually consist of RFID readers with an antenna which communicate with nearby active or passive RFID tags. Due to its technical capabilities, RFID system has been widely deployed as a prevalent technology for numbers of scenarios, e.g., homecare and healthcare systems, industrial automation and asset tracking [15]. Similar to RADAR [16] framework, LANDMARC [17] is also an RFID based positioning scheme. Its superiority to others schemes lies in that its RF map is built by previously placed active tags [18]. By introducing the concept of reference tag, LANDMARC reduces the deployment quantity of RFID readers without compromising its accuracy of localization, and thus can bring down the cost of importing system and can be easily used in the real world.

However, the deployment of extra infrastructure is still costly compared with those modalities based on existing infrastructure. Besides that, on the user's side, they are required to take along the tags, which is not convenient and may bring maintenance issues.

2.1.7 Wireless Local Area Networks (WiFi)

Wireless Local Area Networks (IEEE 802.11 standard, normally called Wi-Fi) have been exploited and employed to predict the location of a target in indoor environ-

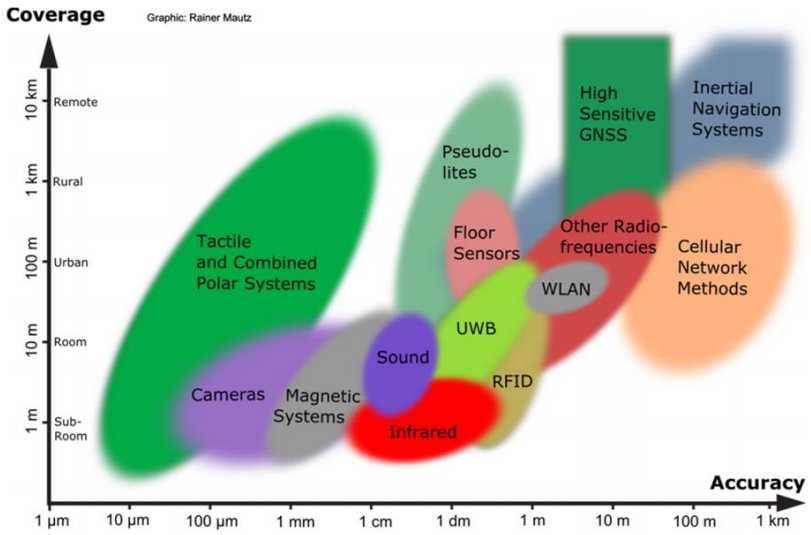

Figure 2.1: Overview of indoor localization technologies in terms of accuracy and coverage [2]

ments for a few years. Since the existing IEEE 802.11 network infrastructures, such as WiFi routers, are widely available in a large number of commercial and residential buildings, and nearly each mobile device now is WiFi enabled [19], it is low-cost and practical to develop a WiFi based IPS to provide LBS in indoor environment.

The localization accuracy of fingerprinting WiFi-based IPS can reach 2m, which largely depends on the density of reference points collected in the offline stage. One well-known WiFi-based IPS is Microsoft research Radar system, which can achieve 2-3 m with a probability of 50% [16]. Ekahau [20], which is a commercial application of WiFi-based IPS, provides services such as estimating tags' positions, testing positioning information, displaying logical areas and analyzing the accuracy of the estimated location and WLAN network design and deployment.

2.2 Algorithms of indoor positioning

The algorithms of indoor positioning can be broadly categorized into two classes, 1) *memoryless localization*: the first class merely relies on an measurement at a given time to compute a position estimate at that time; 2) *tracking*: the second class

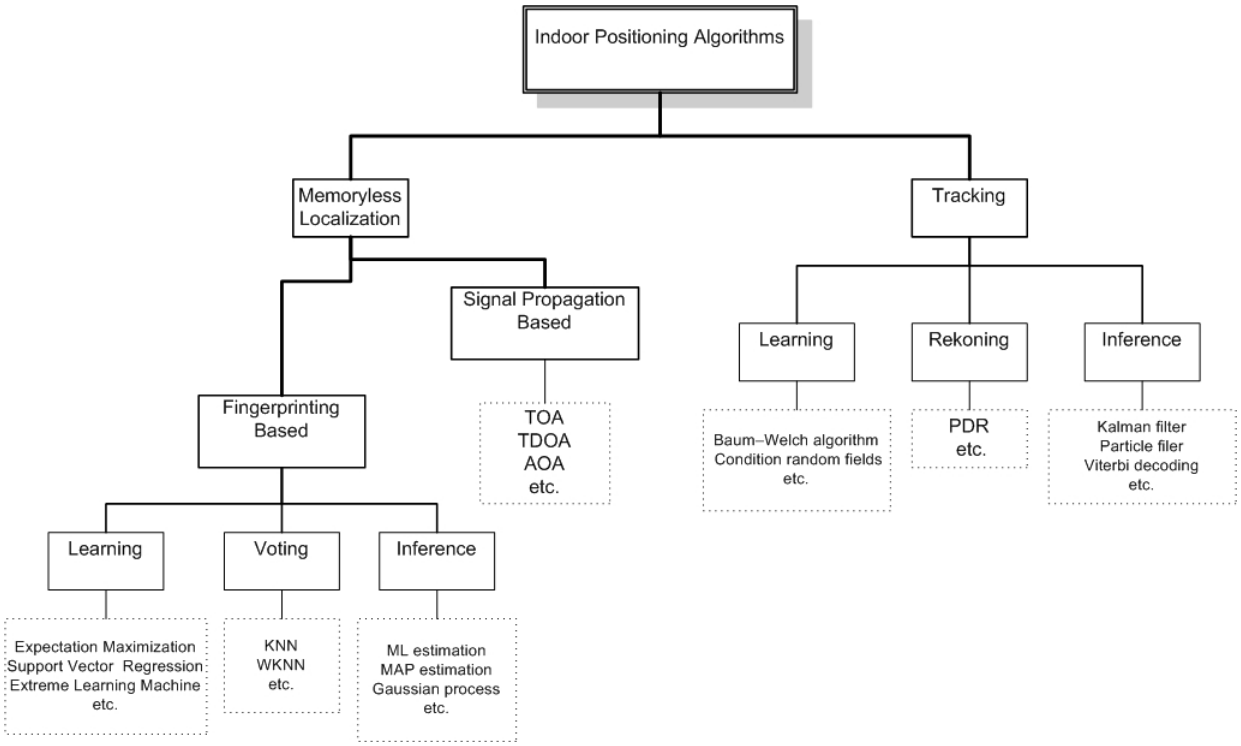

Figure 2.2: The taxonomy of indoor positioning algorithms, where for each class, several representative examples are listed

exploits the correlation in positions observed over time resulting from the physical laws of kinematics. Such algorithms employ the current measurement as well as the measurement history and information from past position estimates, linked by a model of the dynamics of the motion process. For convenience of description, we hence after refer to memoryless localization as *localization*.

The localization approaches have two branches: the signal propagation based and the fingerprinting based. Particularly, the fingerprinting based localization techniques can be further divided into three subclasses: voting, inference and learning.

On the hand, the categories of tracking approaches only include inference and learning. We provide a high-level review of them in the context of an indoor positioning scenario. Figure 2.2 shows the taxonomy of the algorithms, along with specific examples under each class.

In the rest of this section, we will provide a high-level review for them in the in the context of an indoor positioning scenario.

2.2.1 Signal propagation based memoryless localization

The time of signal travelling from the transmitter to the receiver and angle of arrival of a wireless signal have been intensively utilized in the indoor localization problem. Time of arrival (TOA) algorithm uses range (distance) measurements from multiple access points (transmitter) to compute the position of a device (receiver). The distance between the transmitter and receiver is the time of travel multiplied by the speed of light [21]. Angle of arrival (AOA) positioning techniques employ the angle of arrival of a wireless signal to determine the position of a device. Essentially, both TOA and AOA techniques adopt the least squares algorithms to derive the locations.

Denote the position of the receiver and the ith transmitter in two-dimensional Cartesian coordinates as (x, y) and (x_i, y_i), respectively. Further, the distance between the receiver and transmitter is denoted as r_i. The total number of transmitters is assumed as n, and by Pythagoras theorem we have:

$$\mathbf{HX} = \mathbf{B} \tag{2.1}$$

where

$$\mathbf{X} = [x, y]^T \tag{2.2}$$

$$\mathbf{H} = \begin{bmatrix} x_2 - x_1, y_2 - y_1 \\ \vdots \qquad \vdots \\ x_n - x_1, y_n - y_1 \end{bmatrix}, \tag{2.3}$$

and

$$\mathbf{B} = \frac{1}{2} \begin{bmatrix} (r_1^2 - r_2^2) + (x_2^2 + y_2^2) - (x_1^2 + y_1^2) \\ \vdots \\ (r_1^2 - r_n^2) + (x_n^2 + y_n^2) - (x_1^2 + y_1^2) \end{bmatrix}. \tag{2.4}$$

The least square solution of (2.1) is

$$\mathbf{X} = \mathbf{H}^{\dagger}\mathbf{T} \tag{2.5}$$

where $\mathbf{H}^{\dagger}$ is the Moor-Penrose generalized inverse of $\mathbf{H}$.

The angulation counterpart can be derived in a similar manner.The angle between the receiver and transmitter, θ_i can then be written as:

$$tan\theta_i = \frac{y - y_i}{x - x_i} \tag{2.6}$$

Therefore,

$$(x - x_i)sin\theta_i = (y - y_i)cos\theta_i \tag{2.7}$$

Similar to the TOA case, the above defines a system of equations which can also be written in matrix form:

$$\mathbf{HX} = \mathbf{B} \tag{2.8}$$

where,

$$\mathbf{H} = \begin{bmatrix} -sin\theta_1, cos\theta_1 \\ \vdots \qquad \vdots \\ -sin\theta_n, cos\theta_n \end{bmatrix}, \tag{2.9}$$

and

$$\mathbf{B} = \frac{1}{2}\begin{bmatrix} y_1 cos\theta_1 - x_1 sin\theta_1 \\ \vdots \\ y_1 cos\theta_n - x_1 sin\theta_n \end{bmatrix}. \tag{2.10}$$

The solution of (2.8) can also be given by 2.5.

Although TOA, TDOA, and AOA algorithms have been applied broadly in localization scenarios [3, 22–24], their performances deteriorate in indoor environments because of the multi-path effects and the non-LOS (NLOS) condition.

2.2.2 Fingerprinting based memoryless localization

The fingerprinting based approaches to positioning rely on the historical information of signals from a set of reference points with known locations. They involve two

phases: an offline calibration phase and an online localization phase. Site survey of the locations and RSS fingerprints are usually undertaken in the offline stage, whereby a radio map is developed. During the online phase, the location of a device can be estimated by matching its corresponding RSS with those fingerprints stored in the radio map.

Various radio frequency signals can be leveraged as fingerprints. RADAR [16] and Horus [25] utilize WiFi signal, while LANDMARC [17] is based on RFID signal; GSM signals [26], geomagnetism [27] and FM radio [28] can be employed as fingerprints as well for indoor localization [6].

The diversity of the fingerprinting based localization algorithms lie in the different matching techniques. Generally, they can be divided into three classes: voting, inference and learning.

2.2.2.1 Voting based localization

Voting is extensively adopted to aggregate the preferences from multiple reference points to achieve a collective decision. In the context of localization, when a measurement of a target is provided at a given time, the information of reference points can be leveraged to estimate the location at that time by voting rules. To be more specific, denote by $\hat{x}(t)$ as the estimated locations of a target with ground truth $x(t)$ at timestamp t and r_m as the locations of mth reference point out of the M most similar reference points to the target, then $\hat{x}(t)$ can be computed by weighted K nearest neighbor (WKNN) algorithm as follows:

$$\hat{x}(t) = \sum_{m=1}^{M} w_m r_m \tag{2.11}$$

where w_m is the wight based on some measure between $z_m(t)$ (the measurement of mth reference point at timestamp t) and the $z(t)$ (the measurement of the target point at timestamp t). When w_m are set to be the same, WKNN degenerates to K nearest neighbor (KNN).

Voting algorithms are non-parametric and easy to implement, making themselves suitable for numerous scenarios. However, it's tricky to characterize the RSS-position dependency. For simplicity rather than accuracy, prevalent strategies of deriving the weights include the inverse of Euclidean distances of two RSSs [16, 17, 25] and inverse of estimated physical distances between two RSSs by path loss model [29]. It should be pointed out that the denser reference points are, the better performance can be obtained by voting algorithms [21].

2.2.2.2 Inference based localization

The inference-based approaches model the monitored states, the measurements from one or more access points, and the available prior knowledge in a probabilistic model with a set of parameters Θ known a priori. In the context of localization, any temporal correlations between the states are ignored, and only the measurements and priors at any single timestamps are utilized, which means Θ only controls the emission probabilities of measurements given current states. By the Bayes' theorem, the estimated location $\hat{x}(t)$ can be inferred as follows:

$$\hat{x}(t) \propto p(x(t)|\Lambda)p(Z_{1:S}(t)|(x(t)) \tag{2.12}$$

where Λ represents the prior knowledge on $x(t)$, $Z_{1:S}(t)$ denotes the measurements of S reference points at timestamp t.

Many Bayesian inference tools are adopted to predict $\hat{x}(t)$ based on equation (2.12), including maximum likelihood estimation [30], maximum a posteriori estimation [31] and Gaussian process [32]. When inference based localization algorithms incorporate the fingerprinting technique, the measurements of reference points are calibrated in the offline phase and $Z_{1:S}(t)$ degenerate to $Z_{1:S}$.

The emission probabilities of observation given states need to be exactly modeled in advance, thus the performance of inference based localization algorithms largely

depends on the assumption on models and the choice of Θ. Unfortunately, such expert knowledge is unavailable in many scenarios, which is the limitation of inference based localization algorithms when they are applied to real-world applications.

Another shortcoming of inference is its high computational complexity, which incurs poor scalability in large-scale scenarios. When the application doesn't strongly require timeliness, e.g., in the occupancy monitoring cases, inference based algorithms can be adopted for localization.

2.2.2.3 Learning based localization

One difference between learning based approaches and inference based approaches lies in whether to have Θ. In the framework of learning, there are two options to avoid the prior knowledge on Θ or models: 1) *parameter estimation*: Θ are estimated by training the dataset in the first instance, then the refined $\hat{\Theta}$ are further incorporated with the model and remainder procedures are the same with the inference based approaches; 2). *model free*: the second option is similar to the black box in ML, which completely follows a data-driven manner to estimate the locations. It should be noted that our work presented in *Chapters 4–5* falls into the second option, i.e., model-free learning based localization.

Parameter Estimation : The assumed models are iteratively refined by training such that Θ are most consistent with the data observed. Expectation Maximization (EM) approach [33] is widely adopted to derive the maximum likelihood estimate $\hat{\Theta}_{ML}$ or maximum a posteriori estimate $\hat{\Theta}_{MAP}$. EM is an iterative algorithm which alternates between inferring the missing values given the parameters (*E step*), and then optimizing the parameters given the"filled in data (*M step*). Let us define the expected complete data log likelihood in the *ith* iteration as follows:

$$Q(\Theta, \Theta_{i-1}) = E_{x(t)|Z_{1:S}(t),\Lambda,\Theta_{i-1}} \{logp(x(t), Z_{1:S}(t), \Lambda|\Theta)\} \tag{2.13}$$

The *E step* is to compute $Q(\Theta, \Theta_{i-1})$. To derive $\hat{\Theta}_{ML}$, *M step* will optimize $Q(\Theta, \Theta_{i-1})$ with respect to Θ:

$$\Theta_i = arg\max_{\Theta} Q(\Theta, \Theta_{i-1}) \tag{2.14}$$

To compute $\hat{\Theta}_{MAP}$, the *E step* remains the same, and the modified *M step* updates Θ as follows:

$$\Theta_i = arg\max_{\Theta}[Q(\Theta, \Theta_{i-1}) + logp(\Theta)]. \tag{2.15}$$

EM is powerful to estimate parameters and has been heavily utilized in indoor localization [34, 35]. Its rate of convergence on the first few steps is typically quite good, but can become excruciatingly slow when approaching a local optima.

Model Free : Any wrong assumption about the model, e.g., a linear assumption versus the nonlinear ground truth, can limit the performance of a parameter estimation based learning algorithm. An alternative is to use model-free algorithms, e.g., support vector regression (SVR) and ELM, to estimate the parameters. Essentially, non-linear feature mapping (from low dimension to hight dimension) is usually performed beforehand from input space to feature space to improve the representability of data for model-free learning algorithms. As the work in this thesis falls into this category, more comprehensive introductions about ELM and SVR are not presented in this section but in *Chapter 3*. Some related works that use model free approaches can be found in [7, 29, 36].

2.2.3 Tracking

2.2.3.1 Inference based tracking

Tracking assumes that the monitored states are temporally correlated, and estimates the states with all the observed measurements and priors. In practice, the dynamics in states are assumed to be first-order Markovian for simplicity. Compared with the inference approaches for localization, the model parameters Θ in the context of tracking not only controls the emission probability, but also determines

the transition probability from previous states to current state. The estimation can be represented as

$$\hat{x}(t) \propto \underbrace{p(Z_{1:S}(1:t), \Lambda, x(t)|\Theta)}_{A_t} \underbrace{p(Z_{1:S}(t+1:T)|\Lambda, x(t), \Theta)}_{B_t} \tag{2.16}$$

where T is the length of the time-varying sequence $x(1:T)$ and $Z_{1:S}(1:t)$ denotes the measurement sequence of S access points from the beginning to timestamp t. A_t is the joint distribution until time t; B_t is the conditional distribution of the measurements made from timestamp $t+1$ given $x(t)$ and prior knowledge Λ. The Viterbi algorithm ([33, 37]), a dynamic programming algorithm, offers an iterative solution of 2.16. It is similar to the Forward-Backward algorithm used in each iteration, with the subtle difference that it applies a maximization instead of a summing operation in each induction step.

In the case of real-time tracking, B_t is not at hand. With merely A_t, the MMSE position estimate is given as follows:

$$\hat{x}(t) = \int x(t) f(x(t)|Z_{1:S}(1:t)) d(x(t)) \tag{2.17}$$

where the posterior density $f(x(t)|Z_{1:S}(1:t))$ can be estimated recursively via Bayes theorem:

$$f(x(t)|Z_{1:S}(1:t)) = \underbrace{f(x(t)|Z_{1:S}(1:t-1))}_{model\ \ contribution} \underbrace{\frac{f(x(t)|Z_{1:S}(t))}{f(Z_{1:S}(t)|Z_{1:S}(1:t-1))}}_{measurement\ \ conrtibution} \underbrace{\Lambda}_{prior\ \ knowledge} \tag{2.18}$$

With the above formulation, $f(x(t)|Z_{1:S}(1:t))$ can be computed by a *predictor-corrector* structure, i.e., using the model to predict first and then leveraging measurement to correct. The family of Bayesian estimation, such as HMMs, Kalmal and particle filters, follow this structure and have been successfully applied to indoor tracking applications [38–41].

Similar to the case of inference based localization, in the context of tracking, inference based approaches also suffer from the prior knowledge of models. Because

more model parameters (e.g., parameters controlling state transition) are involved in tracking than localization, this problem becomes more severe.

2.2.3.2 Learning based tracking

The algorithm of learning based tracking can also been classified into two categories: the *parameter estimation* based and the *model free* based.

Parameter Estimation : Similar to the one in localization case, EM can be used to estimate Θ at the first instance, but the auxiliary function $Q(\Theta, \Theta_{i-1})$ is replaced with

$$Q(\Theta, \Theta_{i-1}) = E_{x(1:T)|Z_{1:S}(1:T),\Lambda,\Theta_{i-1}} \{logp(x(1:T), Z_{1:S}(1:T), \Lambda|\Theta)\} \tag{2.19}$$

which integrates over state sequences $x(1:T)$. The rest steps are the same as in the learning based localization case.

Model Free : In the sense of model free learning, the outputs of tracking systems are structured in the time horizon. Conventionally, the neighbor positions (say five steps in a sequence) generated by a user are assumed to be smooth. It's notorious for inference based tracking to model the exact conditional probability distributions of observations given states. Kernel based similarity measures are powerful only when suitable kernels are chosen, however this expert knowledge is not always available. One recent work [42] employs the linear chain Conditional Random Fields (CRFs), an undirected graph discriminative model, to directly maximize the conditional probability of states given observations without explicitly specifying the distributions. More importantly, unlike the model based tracking algorithms, where a certain observation is usually assumed to be only determined by its respective states, the data-driven tracking ones extend a single observation to be related with multiple states and for multiple observations to inform a single state. This allows us to capture correlations among observations over time, and to express the extent

to which observations support not only states, but also state transitions. Mathematically,

$$p(X(1:T)|Z_{1:S}(1:T),\Lambda) \propto \prod_{j=1}^{T} \Psi(x(j-1),x(j),Z(1:T),j) \tag{2.20}$$

where Ψ are potential functions composed of V feature functions, each of which is user-defined and reflects in a different way how well the two states are supported by whole observations:

$$\Psi(x(j-1),x(j),Z(1:T),j) = exp(\sum_{i=1}^{V} c_i * f_i(x(j-1),x(j),Z(1:T),j)) \tag{2.21}$$

where c_i are feature weights to be tuned through the offline training.

The only limitation regarding learning based tracking algorithms is their high computation complexity. A large body of recent works are devoted to strategies that can speed up the training process for the cases with numerous states.

2.2.3.3 Reckoning based tracking

Among all the reckoning based tracking algorithms, pedestrian Dead-reckoning (P-DR) is the most affirmatory idea that uses accelerometer and compass to track a target. Based on the accelerometer readings of the mobile phone, it can compute the number of steps a person has walked, and therefore calculate the displacement of the entity. By the compass or the gyroscope readings, the direction of moving can be derived. A tuple $< displacement, direction, time >$ pieces together the human motion vector which can be used to reckon the trajectories of users' movement by a simple linear model. With an accurate starting point of a user, PDR can provide reliable estimation in a short period.

We highlight that basic PDR [1] can only be accurate in the first minutes is due to the errors accrued by the continuous crude measurements. Because of lack of effective

[1]We exclude the modified PDR based approaches, e.g., fusion with landmarks [43] or fingerprinting radio map [42].

means to eliminate accumulative errors, the development of basic PDR studies has reached its bottleneck. Several recent works have been proposed to combine WiFi and IMU sensors on smart phones [40, 43, 44].

2.3 Zonal occupancy monitoring systems

The above sections mainly concern related works about indoor positioning. This section is devoted to the review of zonal occupancy monitoring systems.

Although a number of sensing modalities have been applied to occupancy monitoring systems, WiFi is the most desirable one. Passive infrared (PIR) sensors can detect whether a room is occupied but fail to predict the number of occupants in the room [45]. CO_2 sensors are massively employed to detect the occupant density [46], but the detection agility for variations of CO_2 with occupancy was found to be poor for applications in indoor environments [47, 48].

Some researchers also work on designing more accurate sensors for indoor monitoring tasks, which can realize 30% to 40% energy reduction [49]. Whereas these occupancy based HVAC actuation systems are indeed effective regarding energy saving for HVAC systems, they require deployment of extra monitoring sensors and the design, setup and maintenance of the corresponding data collection scheme [48].

On the other hand, as discussed in Section 2.1.7, two reasons justify WiFi to be the most suitable technique for monitoring systems in commercial and educational buildings: first, the existing IEEE 802.11 network infrastructures, such as WiFi routers, are widely accommodated in massive commercial and residential buildings. Second, nearly every mobile device now is WiFi enabled.

Balaj et al. [48] recently designed an occupancy based HVAC actuation system *Sentiel*, which only uses the Existing WiFi infrastructure. However, *Sentiel* requires all the occupants connecting to certain WLANs to derive the Authentication, Authorization and Accounting (AAA) logs of WiFi clients subsequently. Not only this

approach suffers from the active cooperation (users' registration) of the users, but also the system architecture is too complicated to be applied to real-world HVAC systems. Similarly, Chen et. al [50] proposed an innovative approach that leverages the statistical correlation between wireless connection and energy load variation as a novel tool for occupancy detection and individual consumption monitoring. Nevertheless, their approach is not accurate enough because resolution of the binary information (connected or disconnected) is low.

Hence, there is an urgent need to develop a cheap solution to zonal occupancy monitoring which is free of users' cooperation, lightweight and calibration-free.

2.4 Conclusions

In this chapter, we have surveyed the current major sensing modalities and algorithms of indoor positioning. To accommodate the installation cost, system coverage and accuracy, WiFi is the most suitable modality for indoor positioning. Due to multi-path effect and NLOS transmission channels, a deterministic model of RSS-position relationship is difficult to establish, fingerprinting comes in a data-driven manner to bypass the exact modeling and utilize the power of data for prediction. Thus, the most successful technique for indoor memoryless localization to date is fingerprinting. We also gave a comprehensive review of them based on a taxonomy. We point out that the works in *Chapters 4–6* are to design learning algorithms for WiFi based memoryless localization via fingerprinting.

In the end, related works of zonal occupancy monitoring systems for the optimization and control of HVAC systems were also given, none of which are free of users' cooperation, lightweight and calibration-free.

Chapter 3

Overview of WiFi based Indoor Positioning System and Extreme Learning Machine

In *Chapter 2*, we have reviewed the modern technologies and various algorithms for indoor positioning. Given the above discussion, we found that WiFi based technology is the most suitable candidate for indoor positioning. Hence, they are adopted in the development of IPS. In this chapter, more details regarding them will be provided, serving as preliminaries of *Chapters 4–5*.

3.1 WiFi based Indoor Positioning System

The system architecture of our WiFi based IPS is shown in Figure 3.1. The main components of this system consist of the existing commercial WiFi access points (APs), mobile devices with WiFi function, a backend server and a web-based visualization unit. The operation procedure of our developed system is as follows. Firstly, a data collection App for android devices was developed. After WiFi module of a mobile device is turned on, it can collect RSS information from different

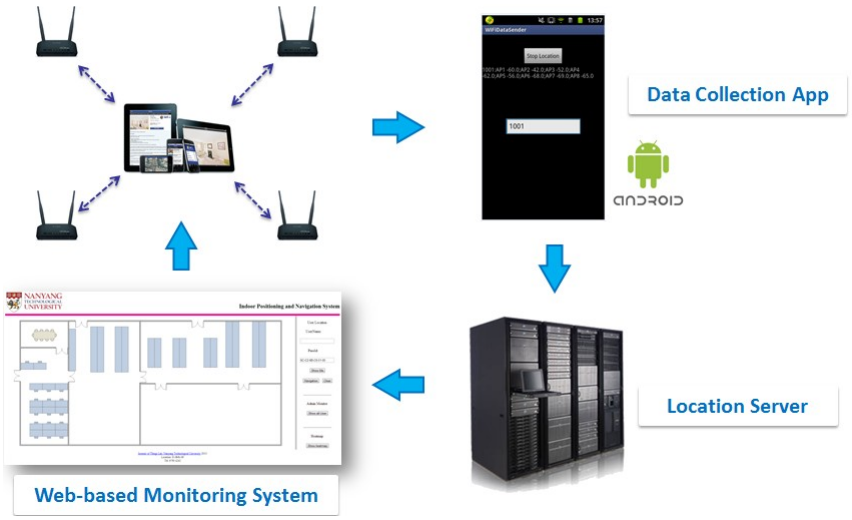

Figure 3.1: System Architecture of our WiFi based IPS

access points every second and sends this information to a location server. The responsibility of the location server is to analyze the RSS, and calculate the estimated position of the mobile device by adopting an appropriate localization algorithm. Then, the user can obtain his or her real time position through our web-based monitoring system directly on his or her mobile devices. Throughout *Chapters 4–5*, we will use the the above developed system in real environment to collect periodical WiFi RSS index for performance evaluation. Figure 3.2 shows the map of our testbed. 36 graduate students and 15 undergraduate students work and study in this lab regularly.

As shown in Table 3.1, the input variable $\mathbf{x}(x_1, x_2, \ldots, x_d)$ is a vector of RSS received from APs in the environment, and $\mathbf{t}(t_1, t_2)$ is the indoor two-dimensional physical coordinates of a target's location. When an AP is undetectable in a position, its corresponding RSS is taken as $-100dBm$.

Essentially, our WiFi based IPS is to approximate the regression model of RSS and locations.

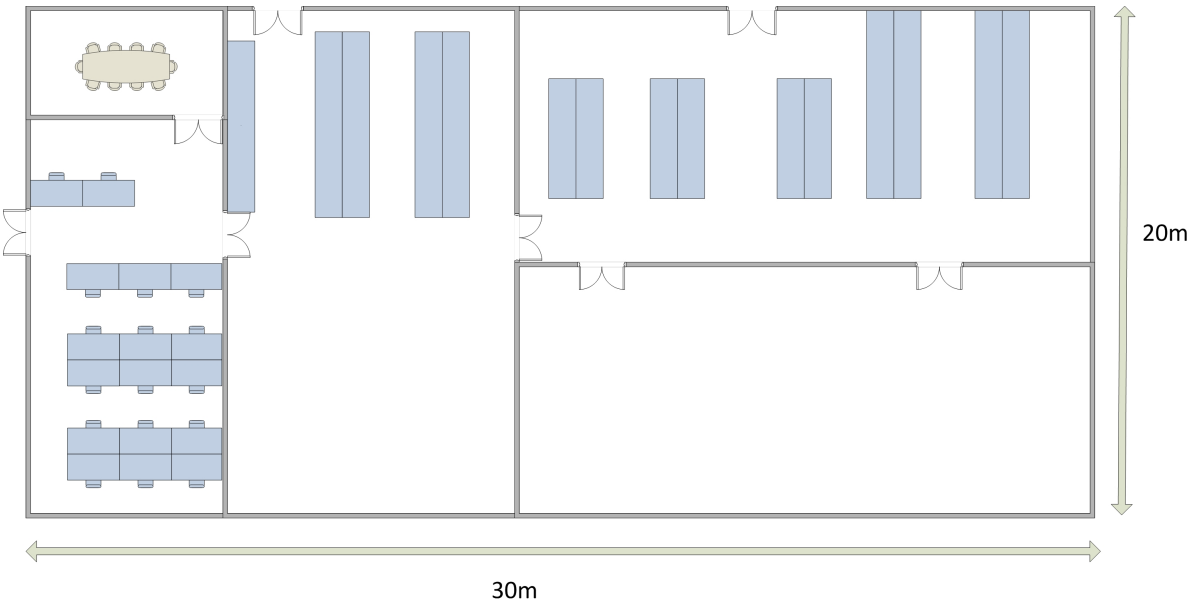

Figure 3.2: Map of the experimental area

Table 3.1: Input variable: RSS ($\mathbf{x}$) & output: location ($\mathbf{t}$)

Location	AP_1	AP_2	AP_3	AP_4	AP_5	AP_6	AP_7
$\mathbf{t}_1$	-53	-87	-73	-80	-79	-87	-89
$\mathbf{t}_2$	-53	-62	-73	-80	-74	-84	-92
$\mathbf{t}_3$	-76	-72	-83	-71	-75	-79	-83
$\vdots$				$\ddots$			
(t_1, t_2)	x_1	x_2	x_3	x_4	x_5	x_6	x_7

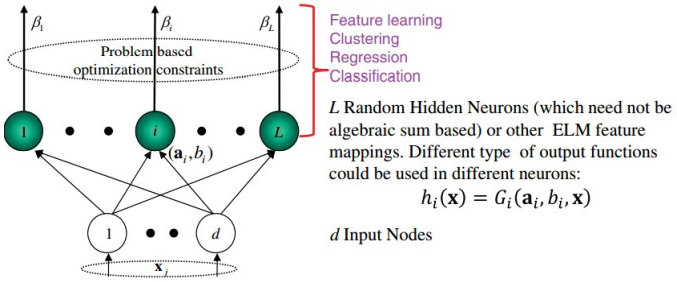

Figure 3.3: ELM Feature Mapping

3.2 Extreme Learning Machine (ELM)

Originally inspired by biological learning to overcome the challenging issues faced by back propagation (BP) learning algorithms, ELM works for the generalize single-hidden layer feedforward networks (SLFNs) while the hidden layer in ELM is free if tuning [51]. ELM has been demonstrated to have good prediction accuracy with low learning complexity [52–54].

Fig. 3.3 shows the structure of ELM [51]. Let $\Upsilon = \{(\mathbf{x}_i, \mathbf{t}_i); i = 1, 2, \ldots, N\}$ be a training set consisting of patterns, where $\mathbf{x}_i \in \mathbf{R}^{1\times d}$ and $\mathbf{t}_i \in \mathbf{R}^{1\times m}$, then the goal of regression is to find the relationship between $\mathbf{x}_i$ and $\mathbf{t}_i$. Since the only parameters to be optimized are the output weights, the training of the basic ELM is equivalent to solving a least squares problem [55].

In the training process, the first stage is that the hidden neurons of ELM map the inputs onto a feature space:

$$\mathbf{h} : \mathbf{x}_i \rightarrow \mathbf{h}(\mathbf{x}_i) \tag{3.1}$$

where $\mathbf{h}(\mathbf{x}_i) \in \mathbf{R}^{1\times L}$.

We denote $\mathbf{H}$ as the hidden layer output matrix (randomized matrix):

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(\mathbf{x}_1) \\ \mathbf{h}(\mathbf{x}_2) \\ \vdots \\ \mathbf{h}(\mathbf{x}_N) \end{bmatrix}_{N\times L} \tag{3.2}$$

with L the dimension of the feature space and $\boldsymbol{\beta} \in \mathbf{R}^{L\times m}$ the output weight matrix that connects the hidden layer with the output layer. Then each output of ELM is given by

$$\mathbf{t}_i = \mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta}, \quad i = 1, 2, \ldots, N \tag{3.3}$$

ELM theory aims to optimize a weighted training error and the norm of output weight [51]:

$$\begin{aligned} \min_{\boldsymbol{\xi},\boldsymbol{\beta}\in\mathbf{R}^{L\times m}} \quad & L_P = \frac{1}{2}\|\boldsymbol{\beta}\|_{\varpi_1}^{\varrho_1} + \frac{C}{2}\sum_{i=1}^{N}\xi_i \\ s.t. \quad & \|\mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} - \mathbf{t}_i\|_{\varpi_2}^{\varrho_2} = \xi_i \qquad i = 1, 2, \ldots, N \end{aligned} \tag{3.4}$$

where $\varrho_1 > 0, \varrho_2 > 0, \varpi_1, \varpi_2 = 0, \frac{1}{2}, 1, 2, \cdots, +\infty$,[1] C is the penalty coefficient on the training errors and $\xi_i \in \mathbf{R}^m$ is the error vector w.r.t. the ith training pattern.

A simplest example of the above is basic ELM [52]:

$$\begin{aligned} \min_{\boldsymbol{\beta}\in\mathbb{R}^{L\times m}} \quad & L_P = \sum_{i=1}^{N}\xi_i \\ s.t. \quad & \|\mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} - \mathbf{t}_i\|^2 = \xi_i \quad i = 1, 2, \ldots, N \end{aligned} \tag{3.5}$$

which can be solved by the least squares method:

$$\boldsymbol{\beta} = \mathbf{H}^{\dagger}\mathbf{T} \tag{3.6}$$

where $\mathbf{H}^{\dagger}$ is the *Moore-Penrose generalized inverse* of $\mathbf{H}$ and $\mathbf{T}$ is a N by m matrix composed of $\mathbf{t}_i$.

Extended from basic ELM, [56] proposed an optimization based ELM (OPT-ELM) for the binary classification problem by introducing inequality constraints. We follow

[1] Unless explicitly specified, $\varpi_1 = \varpi_2 = 2$ for all norm notations.

from [56] to give a form of OPT-ELM for regression problems:

$$\begin{aligned} \min_{\boldsymbol{\xi},\boldsymbol{\beta}\in\mathbf{R}^{L\times m}} \quad & L_P = \frac{1}{2}\|\boldsymbol{\beta}\|^2 + \frac{C}{2}\sum_{i=1}^{N}\xi_i \\ s.t. \quad & \|\mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} - \mathbf{t}_i\| \leq \varepsilon + \xi_i \\ & \xi_i \geq 0 \qquad\qquad i = 1, 2, \ldots, N \end{aligned} \tag{3.7}$$

where ε is a slack variable. This formulation is very similar to support vector regression (SVR) in a nonlinear case [57, 58], which is in the following form:

$$\begin{aligned} \min_{\boldsymbol{\xi},\mathbf{w},b} \quad & L_{P_{SVM}} = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{2}\sum_{i=1}^{N}\xi_i \\ s.t. \quad & \|\mathbf{w}\cdot\phi(\mathbf{x}_i) + b - \mathbf{t}_i\| \leq \varepsilon + \xi_i \\ & \xi_i \geq 0 \qquad\qquad i = 1, 2, \ldots, N \end{aligned} \tag{3.8}$$

where $\phi(\cdot)$ is the nonlinear feature mapping function in support vector regression (SVR), $\mathbf{w}$ is the output weights and b is the approximation (output) bias. ε and ξ_i are as defined in the OPT-ELM case.

Prediction by kernel based methods, e.g., SVR, is of the computational complexity of $\mathcal{O}(Nd)$ since it involves dual variables, while prediction using random-hidden-nodes based methods by primal variables, e.g., ELM is of the computational complexity of $\mathcal{O}(d)$ [33], where d is the number of hidden nodes. In most scenarios, d is far smaller than the number of training samples N, thus ELM and its variants are suitable candidates for solving indoor localization problems due to the real-time requirement for IPSs.

For convenience of description, we henceforth follow from [51] to refer to the formulation of (3.7) as OPT-ELM, while basic ELM stands for the formulation of (3.5). The terminology ELM in the rest of this thesis has more broad meaning, which can be considered as the gathering of basic ELM and its random-hidden-nodes based variants.

3.3 Conclusions

In this chapter, the developed WiFi based IPS and the ELM were introduced to justify our choices and pave the way for *Chapters 4–5*.

Chapter 4

ELM with Dead Zone (DZ-ELM) Based Localization Algorithm

This chapter is concerned with ELM based indoor localization algorithms, taking into account that the measurements of IPSs are disturbed with bounded perturbations. DZ-ELM is proposed and applied into our IPS that can be robust with respect to these uncertainties.

Dead Zone (DZ) approach is very common in control, e.g., parameter estimation in a dynamical system. If an upper bound on the disturbance is known, the adaptation should be stopped when the estimation error is within a selected threshold [59]. DZ based algorithm is superior for its independence of distributions of noises. This non-parametric manner makes the DZ-ELM based IPSs robust to different types of disturbances and flexible in different scenarios. Consistency analyses and a criterion on when to switch the basic ELM to DZ-ELM are further derived. Simulation and real-world experimental results both show that the proposed algorithm can not only provide higher accuracy but also improve the repeatability of IPSs.

4.1 DZ-ELM Algorithm

4.1.1 Problem Formulation

As discussed in Section 3.2, the output layer of ELM can be taken as a linear system with input $\mathbf{H}$. Since $\mathbf{H}$ is the feature space after a non-linear mapping from the input space, if the input data is contaminated, $\mathbf{H}$ is also mixed with disturbances. Thus, it's reasonable that we may add a perturbation term in the input of ELM's output layer. Assume that the perturbation consists of a noise term with even PDF and a bounded deterministic disturbance:

$$\mathbf{T} = \mathbf{H}\boldsymbol{\beta}_0 + \mathbf{e} + \mathbf{d} \tag{4.1}$$

where $\mathbf{e}$ is the noise with even distribution and $\mathbf{d}$ is a deterministic disturbance.

This assumption is reasonable because many types of noises in reality can be seen as a superposition of these two terms. By the introduction of DZ to ELM, we will demonstrate that the effects of noises on the output can be significantly weakened.

Denote $\hat{\boldsymbol{\beta}}$ the estimated output weight and define

$$\varepsilon(\hat{\boldsymbol{\beta}}) = \|\mathbf{T} - \hat{\mathbf{T}}\| = \|\mathbf{H}\tilde{\boldsymbol{\beta}} + \mathbf{e} + \mathbf{d}\| \tag{4.2}$$

where

$$\tilde{\boldsymbol{\beta}} := \boldsymbol{\beta}_0 - \hat{\boldsymbol{\beta}} \tag{4.3}$$

The LS-ELM is about to derive $\hat{\boldsymbol{\beta}}$ by letting

$$\hat{\boldsymbol{\beta}} = arg\min_{\hat{\boldsymbol{\beta}}} \mathbf{E}\{\varepsilon(\hat{\boldsymbol{\beta}})\} \tag{4.4}$$

Let l be a quadratic function which is zero within the interval $[-c, c]$:

$$l(x) = \begin{cases} \frac{1}{2}(x-c)^2 & |x| > c \\ 0 & |x| \le c \end{cases} \tag{4.5}$$

Take $V(\hat{\boldsymbol{\beta}})$ as the loss function(or creiterion) of DZ ELM:

$$V(\hat{\boldsymbol{\beta}}) = \mathbf{E}\{l[\varepsilon(\hat{\boldsymbol{\beta}})]\} \tag{4.6}$$

where $\mathbf{E}\{\cdot\}$ denotes expectation operator and $\varepsilon(\hat{\boldsymbol{\beta}})$ is the prediction error of ELM:

$$\hat{\boldsymbol{\beta}} = arg\min_{\hat{\boldsymbol{\beta}}} \mathbf{E}\{l[\varepsilon(\hat{\boldsymbol{\beta}})]\} \tag{4.7}$$

Hence, the original LS-ELM regression problem is converted to the minimization of loss function $V(\hat{\boldsymbol{\beta}})$. With the assistance of an off-the-shelf convex optimization toolbox [60], this problem can be easily solved once we convert it to a standard convex form.

4.1.2 Consistency Analysis

In some cases, DZ-ELM will be inconsistent. Here the inconsistency implies that the norm of residuals may be zero at all time instants for any output weight values within a set (non-singleton) in the output weight space. Such inconsistency will cause all members of this set indistinguishable. We will discuss this issue in two different scenarios, i.e., the unbounded-noise scenario and bounded-noise scenario.

For the unbounded noise case, Lemma 4.1 concludes that inconsistency won't happen on this scenario. Lemma 4.1 is derived for a scalar case, so $\mathbf{e}$ is substituted with e:

Lemma 4.1. *Let f be an even PDF of the noise e in (4.1). If f is not identically zero outside the interval $[-c, c]$, then $V(\hat{\boldsymbol{\beta}})$ has a global minimum in $\bar{\boldsymbol{\beta}}$, i.e.,*

$$supp(f) \not\subseteq [-c, c] \Rightarrow \hat{\boldsymbol{\beta}} = \bar{\boldsymbol{\beta}} \tag{4.8}$$

where $\mathbf{H}\bar{\boldsymbol{\beta}} = \mathbf{d}$.

When the deterministic perturbation is zero, the global minimum lies in $\boldsymbol{\beta}_0$.

For the bounded noise case, inconsistency may happen. Assume that $\mathbf{e}$ is bounded, e.g. truncated Gaussian noise. If $|\tilde{\boldsymbol{\beta}}| \leq c$, it's evident that a sufficient condition for DZ-ELM to be inconsistent is:

$$|\mathbf{H}_{rmax}\tilde{\boldsymbol{\beta}}| + B_1 \leq c \tag{4.9}$$

where $\tilde{\boldsymbol{\beta}}$ is defined by (4.3), B_1 is an upper bound of the perturbation term $\mathbf{e} + \mathbf{d}$ and $\mathbf{H}_{rmax}$ is the row of $\mathbf{H}$ with the largest row summation.

Lemma 4.2. *Suppose the perturbation and the output of a hidden layer are bounded:*

$$\|\mathbf{e} + \mathbf{d}\|_\infty \leq B_1 < c, \|\mathbf{H}\|_\infty \leq B_2 \tag{4.10}$$

then

$$\Psi \supset \{\hat{\boldsymbol{\beta}} | c > a(\tilde{\boldsymbol{\beta}})B_2 + B_1\} \neq \{\boldsymbol{\beta}_0\} \tag{4.11}$$

where Ψ is a set of $\hat{\boldsymbol{\beta}}$ in which the value of loss function is zero and $a(\tilde{\boldsymbol{\beta}}) = \|\tilde{\boldsymbol{\beta}}\|_1$.

Proof: Holder's inequality gives an upper bound for the first term of (4.9):

$$|\mathbf{H}_{rmax}\tilde{\boldsymbol{\beta}}| \leq \|\mathbf{H}\|_\infty \|\tilde{\boldsymbol{\beta}}\|_1 \leq B_2 a(\tilde{\boldsymbol{\beta}}) \tag{4.12}$$

Since

$$|\varepsilon(\hat{\boldsymbol{\beta}})| \leq B_2 a(\tilde{\boldsymbol{\beta}}) + B_1 \tag{4.13}$$

a DZ width which is strictly grater than B_1 will give an estimate that is inconsistent. The set of parameter vectors in which the value of loss function is zero will be a superset of

$$\{\hat{\boldsymbol{\beta}} | c \geq B_2 a(\tilde{\boldsymbol{\beta}}) + B_1\} \tag{4.14}$$

One can verify that this is the best estimate that holder's inequality can produce.

4.1.3 From LS to DZ

So far we have analysed the applicability of DZ-ELM, however, the tricky part is when to use DZ-ELM instead of the original LS-ELM. Some criterion is to be introduced.

Define

$$\tilde{\boldsymbol{\beta}}_{LS} := \boldsymbol{\beta}_0 - \hat{\boldsymbol{\beta}}_{LS} \tag{4.15}$$

$\hat{\boldsymbol{\beta}}_{DZ}$ defined above can be seen as a function of c. We now denote it as $\hat{\boldsymbol{\beta}}_{DZ}(c)$, where c is the width of the deadzone in (4.5) and redefine $\tilde{\boldsymbol{\beta}}_{DZ}(c)$ as:

$$\tilde{\boldsymbol{\beta}}_{DZ}(c) := \boldsymbol{\beta}_0 - \hat{\boldsymbol{\beta}}_{DZ}(c) \tag{4.16}$$

It's clear that when c is equal to 0, DZ-ELM degenerates to LS-ELM. We continue to compute the derivative of $\tilde{\boldsymbol{\beta}}_{DZ}(c)$ at $c = 0$, and determine the sign of $\tilde{\boldsymbol{\beta}}_{LS}\tilde{\boldsymbol{\beta}}'_{DZ}(0)$. Switching LS based method to DZ based method can provide higher accuracy if and only if the sign is negative. Take the derivative of $\tilde{\boldsymbol{\beta}}_{DZ}(c)$:

$$\frac{d}{dc}\tilde{\boldsymbol{\beta}}_{DZ}(c) = \frac{d}{dc}(\hat{\boldsymbol{\beta}}_{DZ}(c) - \boldsymbol{\beta}_0) = \frac{d}{dc}\hat{\boldsymbol{\beta}}_{DZ}(c) \tag{4.17}$$

How to derive the numerical solution of $\tilde{\boldsymbol{\beta}}'_{DZ}(0)$ is a challenging problem. Here we give a solution when the output of ELM is a scalar case, denoted by t. That is, there is just one sample in the training set. The output of hidden layer is a row vector $\mathbf{h}$, i.e.,

$$\begin{aligned} t &= \mathbf{h}\boldsymbol{\beta} + e + d \\ \hat{t} &= \mathbf{h}\hat{\boldsymbol{\beta}} \\ \varepsilon(\hat{\boldsymbol{\beta}}) &= \mathbf{h}\tilde{\boldsymbol{\beta}} + e + d \end{aligned} \tag{4.18}$$

Steps of computation yield the following equations. Firstly, it follows from [61] that:

$$\frac{d}{d\tilde{\boldsymbol{\beta}}}V(\hat{\boldsymbol{\beta}}) = 0 \Leftrightarrow \mathbf{h}^T(2g - \int_{c-g}^{c+g} F) = 0 \tag{4.19}$$

where

$$\begin{aligned} F(x) &= \int_{-\infty}^{x} f(t)dt \\ g &= \mathbf{h}\tilde{\boldsymbol{\beta}}(c) + d \end{aligned} \tag{4.20}$$

Differentiating both sides of (4.19) w.r.t. c gives

$$0 = \mathbf{h}^T[2\mathbf{h}\tilde{\boldsymbol{\beta}} - F(c+g)(1+\mathbf{h}\tilde{\boldsymbol{\beta}}) + F(c-g)(1-\mathbf{h}\tilde{\boldsymbol{\beta}})] \tag{4.21}$$

Given the fact that f is even,

$$F(-x) = 1 - F(x) \tag{4.22}$$

Hence, at point $c = 0$,

$$\mathbf{h}^T[\mathbf{h}\tilde{\boldsymbol{\beta}}'_{DZ}(0) + 1 - 2F(g)] = 0 \tag{4.23}$$

Furthermore,

$$g|_{c=0} = \tilde{\boldsymbol{\beta}}_{LS}(0) + d \tag{4.24}$$

Substituting this in Eq. (4.23) gives

$$\tilde{\boldsymbol{\beta}}'_{DZ}(0) = (\mathbf{h}^T\mathbf{h})^{-1}[(2F(\mathbf{h}\tilde{\boldsymbol{\beta}}_{LS} + d) - 1)\mathbf{h}^T] \tag{4.25}$$

where straightforward computation results in that:

$$\tilde{\boldsymbol{\beta}}_{LS} = -(\mathbf{h}^T\mathbf{h})^{-1}(\mathbf{h}^T d) \tag{4.26}$$

As shown above, when to introduce DZ to ELM depends only on the sign of $\tilde{\boldsymbol{\beta}}_{LS}\tilde{\boldsymbol{\beta}}'_{DZ}(0)$. It shows that the sign is related to the PDF of noise and the deterministic disturbance rather than the width c. Although the above is merely considering

Table 4.1: Simulation Results of DZ-ELM

Sample sets (Offline + Online)	Training Time (s)	Testing Time (s)	MRSE ± dev. (m)
ELM			
set1	0.219	0.0057	**3.33 ± 0.21**
set2	0.148	0.0082	**2.28 ± 0.23**
set3	0.139	0.0115	**2.36 ± 0.42**
set4	0.139	0.0055	**1.92 ± 0.25**
DZ-ELM			
set1	13.22	0.0084	**2.71 ± 0.11**
set2	6.11	0.0082	**1.81 ± 0.15**
set3	12.03	0.0150	**2.33 ± 0.26**
set4	4.54	0.0065	**1.55 ± 0.19**

a one-training-sample case, it can be extended to the case of multiple training samples as ELM will minimize the estimation errors on average over all samples in that case.

4.2 Performance Evaluation

4.2.1 Simulation Results

In this section, simulations of different types of disturbances are conducted with DZ-ELM and LS-ELM. An issue of ELM is the variation of repeated realizations, namely, ELM with same parameters setting, e.g., number of hidden units, of the same training set may draw different results. This lack of repeatability is caused by two reasons. One reason is that the number of hidden units is not infinite so that the universal approximation using SLFNs with random nodes may not be accurate [53]. The other reason is that we can't ignore the disturbance in both training data and testing data. Namely we can't universally approximate a certain function with noises. We measure this kind of unrepeatable property by the deviation of repeated realizations' results.

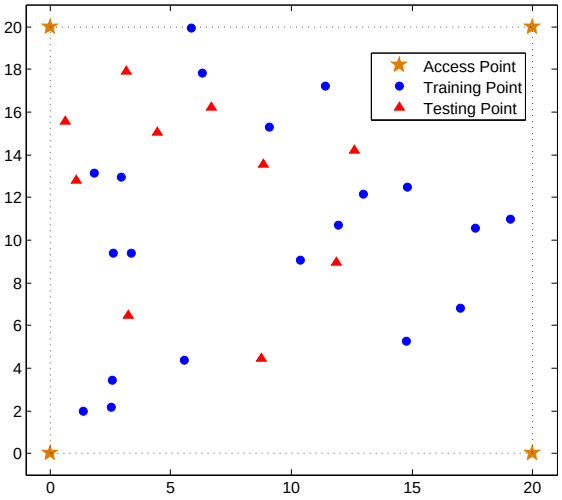

Figure 4.1: Positions of the WiFi access points, offline calibration points and online testing points in the simulated field

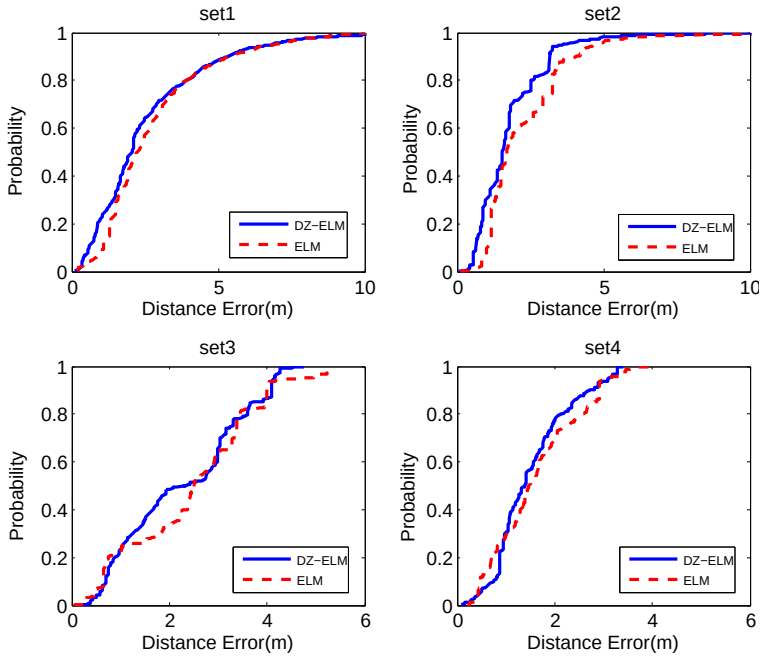

Figure 4.2: Cumulative percentile of error distance for different data sets

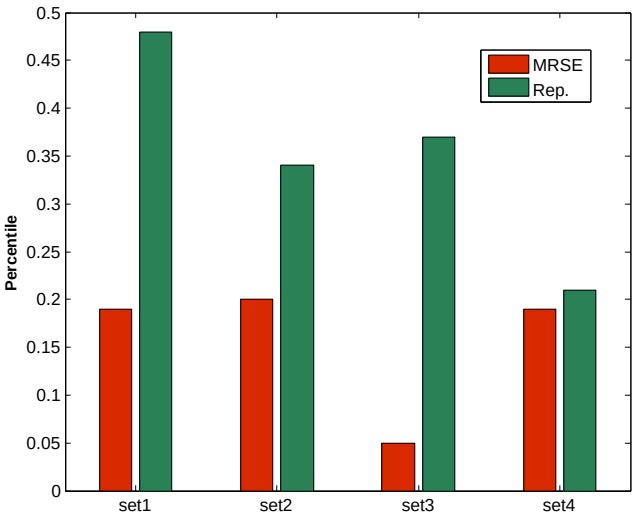

Figure 4.3: Performance improvement of DZ-ELM

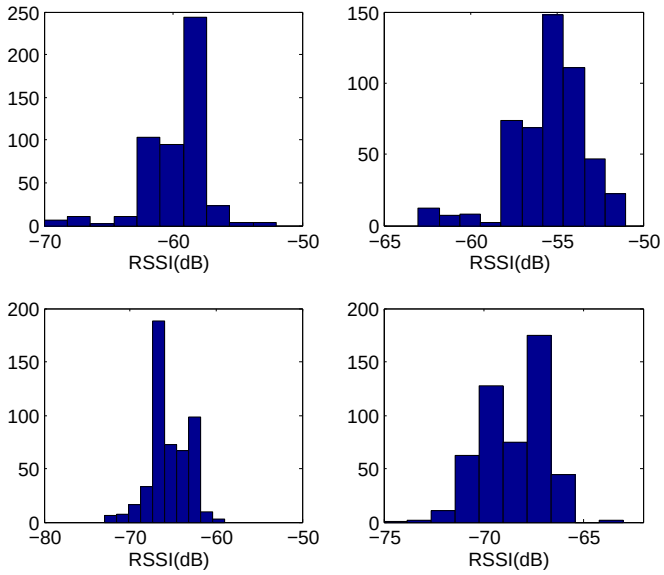

Figure 4.4: RSSI distribution of four APs at one position

We develop a simulation environment using Matlab R2013a in order to evaluate the performance of the proposed DZ-ELM approach before any real-world experiment is conducted. As shown in Figure 4.1, we assume a $20m \times 20m$ room where 4 WiFi access points are installed at the four corners of the room. The indoor path loss model [62], which provides a relation between the total path loss PL (dBm) and distance d (m), is employed to generate the synthetic data.

$$PL(d) = PL_0 - 10\alpha log(d) + X_\sigma \tag{4.27}$$

where PL_0 is the path loss coefficient and it is set to be -40dBm in our simulation. X_σ represents a random noise, and α is the path loss exponent.

Four different types of data with perturbations are generated, i.e., data with Gaussian noise and deterministic sine disturbance, data with environmental dynamics (path loss exponent changed), data with zero-mean Gaussian noise, and data with outliers are generated respectively to test the robustness of DZ-ELM.

The accuracy is measured by the mean root square error (MRSE) over r repeated realizations. Beside the accuracy, we also take Repetition (REP) into account, which is measured by the deviation of the MRSE over the repeated realizations. REP is proposed based on the fact that ELM with same parameters, e.g., the number of hidden nodes, in the same training data set may draw quite different results. In our case r is taken as 50.

$$\begin{aligned} MRSE &= \frac{1}{r}\sum_{j=1}^{r}\left(\frac{1}{s}\sum_{i=1}^{s}\left\|\mathbf{t}_i - \mathbf{h}_i\hat{\boldsymbol{\beta}}\right\|\right)_j \\ REP &= \sqrt{\frac{1}{r}\sum_{j=1}^{r}\left(\frac{1}{s}\sum_{i=1}^{s}\left\|\mathbf{t}_i - \mathbf{h}_i\hat{\boldsymbol{\beta}}\right\| - MRSE\right)_j^2} \end{aligned} \tag{4.28}$$

where s is the number of samples. Table 5.1 shows a comparison between DZ-ELM and basic ELM. Set1,2,3,4 respectively represent data set with Gaussian noise and deterministic sine disturbance, data set with environmental dynamics (path loss exponent changed), data set with zero-mean Gaussian noise, and data set with outliers.

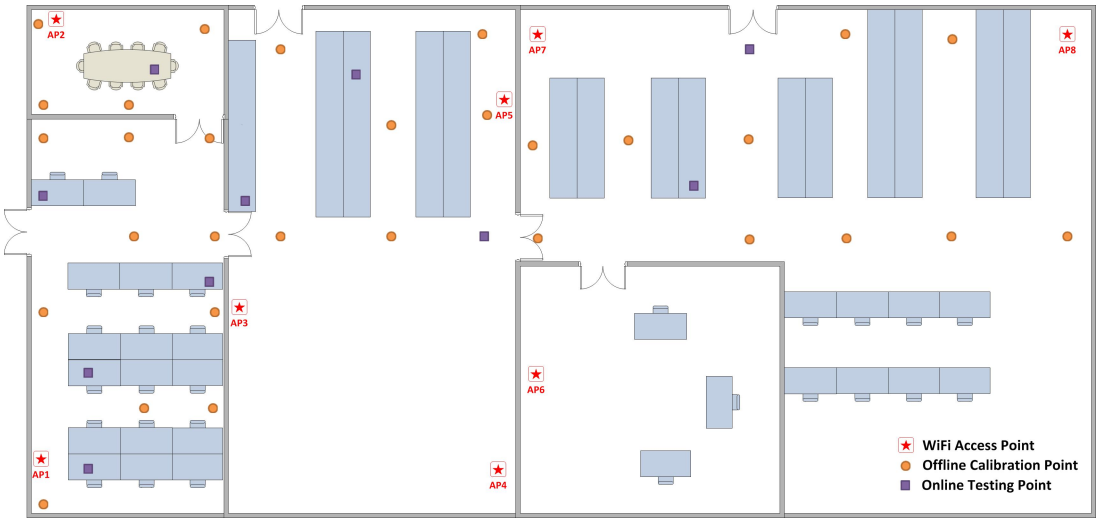

Figure 4.5: Positions of the WiFi access points, offline calibration points and online testing points in the test-bed

It shows that DZ-ELM enhanced not only the accuracy but the repeatability of IPS. Specific performance improvement is illustrated in Figure 4.3. The comparison of two cumulative percentile of error distance is presented in Figure 4.2. It has been shown that DZ-ELM has obvious performance enhancement, about 20% for almost all the data sets.

4.2.2 Evaluation in Real-world IPSs

We conducted extensive experiments to evaluate the performance of the proposed DZ-ELM approach. The testbed is the Internet of Things Laboratory as described in Section 3.1 and illustrated in Fig. 3.1 .

The placement of APs and locations of reference and testing points are shown in Figure 4.5. 8 D-Link DIR-605L WiFi Cloud Routers are utilized as WiFi access points for our experiments. In order to collect WiFi RSS fingerprints from multiple access points simultaneously, one android application which can collect data once per second is developed. This Android application is installed on a Samsung I929 Galaxy SII mobile phone. All the WiFi RSS fingerprints at offline calibration points, online calibration points and online testing points are collected using this phone for performance evaluation.

The initial DZ-ELM model was built up by the following steps. During the offline phase, 30 offline calibration points were selected and 500 WiFi RSS fingerprints were collected at each point. The positions of these 30 offline calibration points are demonstrated in Figure 4.5. By leveraging these 15000 WiFi RSS fingerprints and their physical positions as training inputs and training targets accordingly, the DZ-ELM model was constructed. During the online phase, we continued to collect WiFi RSS fingerprints at several online calibration points and online testing points for five days. In each day, two distinct online calibration points and two distinct online testing points were selected in order to reflect the environmental dynamics. The positions of these total 10 online calibration points and 10 online testing points are also presented in Figure 4.5. 500 WiFi RSS fingerprints are collected at each point.

Unlike [29], which gives little attention to the property of signal strength received by APs, we firstly investigate the received signal strength indication(RSSI) of APs. As illustrated in Figure 4.4, the noise distribution can not be simply modelled as normal distribution, and RSSIs are contaminated by some outliers. Thus, typical LS approach based ELM can't work well on this situation, which motivates the adoption of DZ-ELM.

Based on the analysis in Subsection 3.2, the type of activation function, width of dead zone and the number of hidden nodes in the DZ-ELM hidden layer are three key parameters affecting the performance of DZ-ELM during the initialization phase, which is accordingly the offline calibration phase in our case. By using the 15000 WiFi RSS fingerprints we collected during the offline calibration phase, we evaluate the performance of three different activation functions: radial basis function (RBF) $G(\mathbf{a}, b, \mathbf{x}) = e^{-b||\mathbf{x}-\mathbf{a}||^2}$, sine function $G(\mathbf{a}, b, \mathbf{x}) = sin(\mathbf{a}^T\mathbf{x}+b)$ and hard-limit transfer (hardlim) function $G(\mathbf{a}, b, \mathbf{x}) = hardlim(\mathbf{a}^T\mathbf{x} + b)$ with different numbers of hidden nodes. Based on our experiment analysis, when the hardlim function is chosen as the activation function, the number of hidden layers is set to 600 and the width of DZ is selected to be 2.0, DZ-ELM has the best performance. We also picked the best performance case of ELM and conducted the comparison. As shown in Table 4.2,

Table 4.2: Comparison of ELM and DZ-ELM in IPS

	Training Time (s)	Testing Time (s)	MRSE (m)	Rep. (m)
ELM	2.51	0.095	2.84	0.33
DZ-ELM	7.14	0.099	2.19	0.16

it turns out that DZ-ELM can provide localization accuracy of 2.19m while basic ELM only achieves an accuracy of 2.84m, but DZ-ELM is more time-consuming. The cumulative percentile of error distance of the two methods are shown in Fig. 5.3. The intersection of the two curves is interpreted as follows. The LS approach is to minimize the square root of distance errors. On the other hand, DZ-ELM aims to confine as many distance error's into the DZ region as possible. However, there are no fixed procedures to determine which method is better, their performances are subjected to the types of noises and outliers. This explains well the intersection of two curves in Fig. 5.3. The LS based ELM, has more values near zero but DZ-ELM confines more values into a certain interval.

4.3 Conclusion

In this chapter, we proposed a new ELM algorithm, DZ-ELM, to enhance the robustness of ELM. Consistency and criterion on when DZ-ELM should be applied were studied from theoretical aspects. Simulations based on real setting with different types of perturbations were conducted to compare the performance of ELM and DZ-ELM. It turns out that DZ-ELM can improve both the accuracy and the repeatability of IPS under all types of perturbations. We noted that the computational cost of DZ-ELM is higher than LS-ELM. However, the slightly higher offline training time is not a concern. Experiments were carried out which demonstrated that the proposed DZ-ELM reduces effects of outliers, environmental dynamics and noises with uneven distribution.

In summary, DZ-ELM can provide higher accuracy and better repeatability than ELM due to the introduction of the dead zone. However, this work only considers

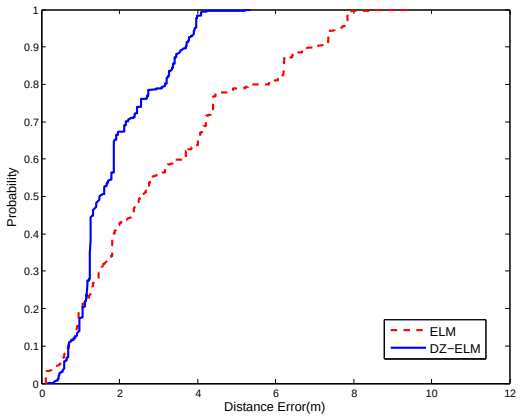

Figure 4.6: Cumulative percentile of error distance for IPS testing results

the uncertainties in input data. In the next chapter, we will study uncertainties in both input and output data.

Chapter 5

Robust Extreme Learning Machine (RELM) Based Localization Algorithm

The uncertainties of IPSs have been studied in *Chapters 4*. DZ-ELM is capable of alleviating the problem of disturbances but it's sensitive to the width c of DZ and can be inconsistent when the c is not chosen properly. This chapter is concerned with a more general case that noises lie in both inputs and outputs.The second order cone programming, a technique of robust optimization, is adopted to handle uncertainties.

Noisy measurements are inevitable, considering that manual observational errors of calibrated points happen throughout the calibration phase of a fingerprinting approach. In addition, signal variation and ambient dynamics also affect the signals received by APs. These adverse factors can be considered as the uncertainties lying in inputs and outputs of an IPS.

Conventional machine learning algorithms, e.g. SVR and BP neural network, are inherently time-consuming, thus it would aggravate the situation of slow training rate for them if we keep optimizing these algorithms. Alternatively, many researchers

bypass complicated machine learning methods, but think of other ways to improve the KNN based IPS. Kothari et al. [63] utilized the combination of complementary localization algorithms of dead reckoning and WiFi signal strength fingerprinting to achieve robust indoor localization, nevertheless, a disadvantage of dead reckoning is that the errors of the process are cumulative, since new positions are calculated solely from previous positions. Meng et al. [64] also proposed a robust non-iterative three-step location sensing method, but its capability of reducing worst-case error and variance is comparatively limited. Other robust indoor localization algorithms may demand extra infrastructure or users' interaction during calibration phases, which is not cost-efficient in reality.

In this chapter, we propose two RELMs to boost the robustness of IPSs. It should be noted that, RELMs inherit the training and testing speed of ELM, though it's comparatively slower than basic ELM and OPT-ELM, it's still competitive with conventional ML algorithms based IPSs.

5.1 RELM Algorithm

5.1.1 Uncertainties of Input and Output Data

RELM is proposed under a probabilistic framework. Assume that both input $\mathbf{x}$ and output data $\mathbf{t}$ are perturbed by noises. Since $\mathbf{H}$ is the feature space after non-linear mapping from the input space, if the input data is contaminated, $\mathbf{H}$ is also mixed with disturbances. We follow from [65] to assume that the disturbances in the feature space are additive.

$$\begin{aligned} \mathbf{h}(\mathbf{x}_i) &= \mathbf{h}(\mathbf{x}_i)_{true} + (\boldsymbol{\iota}_1)_i \\ \mathbf{t}_i &= (\mathbf{t}_i)_{true} + (\boldsymbol{\iota}_2)_i \end{aligned} \tag{5.1}$$

where $(\boldsymbol{\iota}_1)_i$ and $(\boldsymbol{\iota}_2)_i$ are uncorrelated perturbations in the feature space and output space with proper dimensions respectively. The new vector $\mathbf{y}_i \in \mathbf{R}^{1\times(L+m)}$ is the

ith input and ith output observation, i.e., $\mathbf{y}_i = [\mathbf{h}(\mathbf{x}_i), \mathbf{t}_i]$. And now we give the following definitions:

$$\begin{aligned} &\bar{\mathbf{h}}(\mathbf{x}_i) = \mathbf{E}(\mathbf{h}(\mathbf{x}_i)), \qquad &&\bar{\mathbf{t}}_i = \mathbf{E}(\mathbf{t}_i) \\ &\Sigma^i_{hh} = \mathbf{Cov}(\mathbf{h}(\mathbf{x}_i), \mathbf{h}(\mathbf{x}_i)), &&\Sigma^i_{tt} = \mathbf{Cov}(\mathbf{t}_i, \mathbf{t}_i) \end{aligned} \tag{5.2}$$

where $\mathbf{E}(\cdot)$ and $\mathbf{Cov}(\cdot)$ represent expectation and covariance operators for random variables separately. Since the noises in the feature space $(\boldsymbol{\iota}_1)_i$ and output space $(\boldsymbol{\iota}_2)_i$ are uncorrelated, i.e., $\Sigma^i_{ht} = 0$, we have

$$\begin{aligned} &\bar{\mathbf{y}}_i = \mathbf{E}([\mathbf{h}(\mathbf{x}_i), \mathbf{t}_i]) = [\bar{\mathbf{h}}(\mathbf{x}_i), \bar{\mathbf{t}}_i] \\ &\Sigma^i_{yy} = \mathbf{Cov}(\mathbf{y}_i, \mathbf{y}_i) = \begin{bmatrix} \Sigma^i_{hh} & 0 \\ 0 & \Sigma^i_{tt} \end{bmatrix}_{(L+m)\times(L+m)} \end{aligned} \tag{5.3}$$

The ith prediction error is denoted by $\mathbf{e}_i \in \mathbf{R}^{1\times m}$ and its expectation $\bar{\mathbf{e}}_i$ is defined as follows:

$$\mathbf{e}_i = \mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} - \mathbf{t}_i, \bar{\mathbf{e}}_i = \bar{\mathbf{h}}(\mathbf{x}_i)\boldsymbol{\beta} - \bar{\mathbf{t}}_i \tag{5.4}$$

It follows from [65] and [66] that, by inserting close to mean (CTM) and small residual (SR) constraints into SVR, the predictions can be robust to perturbations in the data set.

CTM is a criterion on that we require the prediction errors to be insensitive to the distribution of the noises in input and output data:

$$\underset{x_i, y_i}{Pr}\{|e_i - \bar{e}_i| \geq \theta_i\} \leq \eta \quad i = 1, 2, \ldots, N \tag{5.5}$$

x_i, y_i here are the input and output data, and θ_i means the confidence threshold while η denotes the maximum tolerance of the deviation.

An alternative way to boost the robustness is to restrict the residual to be small, which leads to the small residual (SR) constraint,

$$\underset{x_i,y_i}{Pr}\{|e_i| \geq \xi_i + \varepsilon\} \leq \eta \tag{5.6}$$

where ξ_i corresponds to the prediction error and ε is a slack variable. Compared with the CTM constraint, the SR constraint requires the estimator to be robust in terms of deviations which lead to larger estimation error rather than centering. In fact, both CTM and SR constraints are robust constraints utilized to bound probabilities of highly deviated errors subject to second order moment constraints.

5.1.2 Sufficient Condition of CTM constraint

It should be pointed out that, the above two robust constraints only consider a scalar output case, however, the outputs of IPSs are usually vectors. Therefore different constraints should be provided for our problem setting. We now give our CTM constraint:

$$\underset{\mathbf{h}(\mathbf{x}_i),\mathbf{t}_i}{Pr}\{\|\mathbf{e}_i - \bar{\mathbf{e}}_i\|^2 \geq \theta_i^2\} \leq \tau \quad i = 1, 2, \ldots, N \tag{5.7}$$

where θ_i is still a confidence threshold and τ stands for some probability. Nevertheless, CTM constraints in this form are intractable. Multidimensional Chebyshev's inequality is leveraged to convert the constraints into tractable ones.

Lemma 5.1. *[67] Let* $\mathbf{z}$ *be an* m*-dimensional random row vector with expected value* $\bar{\mathbf{z}}$ *and positive-definite covariance* Σ*, then*

$$Pr\{(\mathbf{z} - \bar{\mathbf{z}})\Sigma^{-1}(\mathbf{z} - \bar{\mathbf{z}})^T \geq \theta^2\} \leq \frac{m}{\theta^2} \tag{5.8}$$

Proposition 5.1. *For* $\mathbf{z}$ *and* Σ *defined in Lemma 5.1, if* $\|\mathbf{z}\|^2 \geq \epsilon\|\Sigma\|$*, then* $\mathbf{z}\Sigma^{-1}\mathbf{z}^T \geq \epsilon$*.*

Proof. Since Σ is a real-valued symmetric matrix, it can be diagonalized as $\Sigma = P^{-1}\Lambda P$. Λ here is a real-valued matrix with eigenvalues of Σ on its diagonal. It can be shown that

$$\Lambda \leq \|\Sigma\|\mathbf{I} \Rightarrow \Lambda^{-1} \geq \|\Sigma\|^{-1}\mathbf{I} \tag{5.9}$$

which leads to

$$\begin{aligned} \mathbf{z}\Sigma^{-1}\mathbf{z}^T &= \mathbf{z}P^{-1}\Lambda^{-1}P\mathbf{z}^T \\ &\geq \frac{\mathbf{z}\mathbf{z}^T}{\|\Sigma\|} \end{aligned} \tag{5.10}$$

and (5.10) gives rise to

$$\|\mathbf{z}\|^2 \geq \epsilon\|\Sigma\| \Rightarrow \mathbf{z}\Sigma^{-1}\mathbf{z}^T \geq \epsilon \tag{5.11}$$

■

Proposition 5.1 also implies

$$Pr\{\|\mathbf{z}\|^2 \geq \epsilon\|\Sigma\|\} \leq Pr\{\mathbf{z}\Sigma^{-1}\mathbf{z}^T \geq \epsilon\} \tag{5.12}$$

Theorem 5.1. *Let $\boldsymbol{\beta} \in \mathbf{R}^{L\times m}$ and $\boldsymbol{\omega} = [\boldsymbol{\beta}^T, -\mathbf{1}]^T \in \mathbf{R}^{(L+m)\times m}$ and Σ^i_{yy} as defined in (5.3), then a sufficient condition for (5.7) is*

$$\| \left(\Sigma^i_{yy}\right)^{\frac{1}{2}} \boldsymbol{\omega}\| \leq \theta_i\sqrt{\tau/m} \tag{5.13}$$

where $-\mathbf{1}$ is a vector of all entries of -1 with proper length.

Proof. Substitute $\mathbf{e}_i, \theta_i$ for $\mathbf{z}, \theta$ into (5.8), we have

$$\underset{\mathbf{h}(\mathbf{x}_i),\mathbf{t}_i}{Pr}\left\{(\mathbf{e}_i - \bar{\mathbf{e}}_i)\left(\Sigma^i_{ee}\right)^{-1}(\mathbf{e}_i - \bar{\mathbf{e}}_i)^T \geq \theta_i^2\right\} \leq \frac{m}{\theta_i^2} \tag{5.14}$$

which together with (5.12), leads to

$$\begin{aligned}
&\underset{\mathbf{h}(\mathbf{x}_i),\mathbf{t}_i}{Pr}\{\|\mathbf{e}_i-\bar{\mathbf{e}}_i\|^2\geq\theta_i^2\}\\
\leq&\underset{\mathbf{h}(\mathbf{x}_i),\mathbf{t}_i}{Pr}\{(\mathbf{e}_i-\bar{\mathbf{e}}_i)\left(\Sigma_{ee}^i\right)^{-1}(\mathbf{e}_i-\bar{\mathbf{e}}_i)^T\geq\frac{\theta_i^2}{\|\Sigma_{ee}^i\|}\}\\
\leq&\frac{m\|\Sigma_{ee}^i\|}{\theta_i^2}
\end{aligned}\tag{5.15}$$

Thus $\frac{m\|\Sigma_{ee}^i\|}{\theta_i^2}\leq\tau$ is a sufficient condition for (5.7). By taking into account that

$$\Sigma_{ee}^i=\boldsymbol{\omega}^T\Sigma_{yy}^i\boldsymbol{\omega},\tag{5.16}$$

inserting (5.16) into $\frac{m\|\Sigma_{ee}^i\|}{\theta_i^2}\leq\tau$ and then taking the square root on both sides, (5.13) follows. ■

5.1.3 Sufficient Condition of SR constraint

A sufficient condition of SR constraint can be derived in the same fashion. The SR constraint in our case is

$$\underset{\mathbf{h}(\mathbf{x}_i),\mathbf{t}_i}{Pr}\{\|\mathbf{e}_i\|^2\geq(\xi_i+\varepsilon)^2\}\leq\tau\quad i=1,2,\ldots,N\tag{5.17}$$

Theorem 5.2. *Let $\boldsymbol{\beta}\in\mathbf{R}^{L\times m}$, $\boldsymbol{\omega}=[\boldsymbol{\beta}^T,-\mathbf{1}]^T\in\mathbf{R}^{(L+m)\times m}$ and Σ_{yy}^i as defined in (5.3), then a sufficient condition for (5.17) is*

$$\left\|\begin{array}{c}\left(\Sigma_{yy}^i\right)^{\frac{1}{2}}\boldsymbol{\omega}\\ \bar{\mathbf{h}}(\mathbf{x}_i)\boldsymbol{\beta}-\bar{\mathbf{t}}_i\end{array}\right\|\leq(\xi_i+\varepsilon)\sqrt{\tau/m}\tag{5.18}$$

where $-\mathbf{1}$ is a vector of all entries of -1 with proper length.

Proof. Taking $\mathbf{e}_i\mathbf{e}_i^T \in \mathbf{R}$ as a random variable, from Markov's inequality, we have

$$\begin{aligned}\Pr_{\mathbf{h}(\mathbf{x}_i),\mathbf{t}_i}\{\|\mathbf{e}_i\|^2 \geq (\xi_i+\varepsilon)^2\} &= \Pr_{\mathbf{h}(\mathbf{x}_i),\mathbf{t}_i}\{\mathbf{e}_i\mathbf{e}_i^T \geq (\xi_i+\varepsilon)^2\} \\ &\leq \frac{\mathbf{E}(\mathbf{e}_i\mathbf{e}_i^T)}{(\xi_i+\varepsilon)^2}\end{aligned}$$

Denote $tr(\cdot)$ as the trace operator of a matrix,

$$\begin{aligned}\mathbf{E}(\mathbf{e}_i\mathbf{e}_i^T) &= \mathbf{E}\{tr(\mathbf{e}_i^T\mathbf{e}_i)\} \\ &= \mathbf{E}\left\{tr(\mathbf{e}_i^T\mathbf{e}_i - \bar{\mathbf{e}}_i^T\bar{\mathbf{e}}_i)\right\} + tr(\bar{\mathbf{e}}_i^T\bar{\mathbf{e}}_i) \\ &= tr(\Sigma_{ee}^i + \bar{\mathbf{e}}_i^T\bar{\mathbf{e}}_i)\end{aligned} \tag{5.19}$$

Since Σ_{ee}^i and $\bar{\mathbf{e}}_i^T\bar{\mathbf{e}}_i$ are both positive semi-definite, which implies that $\Sigma_{ee}^i + \bar{\mathbf{e}}_i^T\bar{\mathbf{e}}_i$ is positive semi-definite. Since

$$\|\Sigma_{ee}^i + \bar{\mathbf{e}}_i^T\bar{\mathbf{e}}_i\| = \max\{\lambda_1, \ldots, \lambda_m\} \tag{5.20}$$

where λ_i stands for an eigenvalue of $\Sigma_{ee}^i + \bar{\mathbf{e}}_i^T\bar{\mathbf{e}}_i$, we have

$$tr(\Sigma_{ee}^i + \bar{\mathbf{e}}_i^T\bar{\mathbf{e}}_i) \leq m\|\Sigma_{ee}^i + \bar{\mathbf{e}}_i^T\bar{\mathbf{e}}_i\| \tag{5.21}$$

which leads to

$$m\|\Sigma_{ee}^i + \bar{\mathbf{e}}_i^T\bar{\mathbf{e}}_i\| = m\left\|\begin{array}{c}\left(\Sigma_{yy}^i\right)^{\frac{1}{2}}\boldsymbol{\omega} \\ \bar{\mathbf{h}}(\mathbf{x}_i)\boldsymbol{\beta} - \bar{\mathbf{t}}_i\end{array}\right\|^2 \tag{5.22}$$

By letting

$$\frac{m}{(\xi_i+\varepsilon)^2}\left\|\begin{array}{c}\left(\Sigma_{yy}^i\right)^{\frac{1}{2}}\boldsymbol{\omega} \\ \bar{\mathbf{h}}(\mathbf{x}_i)\boldsymbol{\beta} - \bar{\mathbf{t}}_i\end{array}\right\|^2 \leq \tau \tag{5.23}$$

and taking square root on both sides, it's clear that (5.18) is a sufficient condition for (5.17). ∎

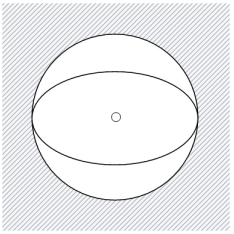

Figure 5.1: Shadow area indicates the possible region the random variable may fall into

5.1.4 Geometric Interpretation

1. Theorem 5.1 can be interpreted as that the chance of a random variable lying outside a sphere with radius $\sqrt{\epsilon\|\Sigma\|}$ is greater than that of a random variable lying outside an ellipsoid with radius $\sqrt{\epsilon}$ and covariance matrix Σ. This is intuitive because the largest length of semi-axe of the ellipsoid is equal to the radius of the sphere and they share the same center. Fig. 5.1 gives an illustration when the ellipsoid and sphere are projected onto a 2-D space.

2. The above CTM robust criterion can be understood as a restriction that each training data $\mathbf{y}_i$ picked from the ellipsoid $\Psi_i(\bar{\mathbf{y}}_i, \Sigma^i_{yy}, \frac{m}{\tau}^{\frac{1}{2}})$ satisfies the inequality

$$\|\mathbf{e}_i - \bar{\mathbf{e}}_i\| \leq \theta_i \tag{5.24}$$

where

$$\begin{aligned} &\Psi_i(\bar{\mathbf{y}}_i, \Sigma^i_{yy}, \sqrt{\frac{m}{\tau}}) \\ \stackrel{\Delta}{=}&\{\mathbf{y}_i | (\mathbf{y}_i - \bar{\mathbf{y}}_i)(\Sigma^i_{yy})^{-1}(\mathbf{y}_i - \bar{\mathbf{y}}_i)^T \leq \frac{m}{\tau}\} \end{aligned} \tag{5.25}$$

From Theorem 5.1, we have

$$\sqrt{m/\tau}\|{\Sigma^i_{yy}}^{\frac{1}{2}}\boldsymbol{\omega}\| \leq \theta_i \tag{5.26}$$

Further, by noting that

$$\begin{aligned}\|\mathbf{e}_i - \bar{\mathbf{e}}_i\| &= \|(y_i - \bar{y}_i)\boldsymbol{\omega}\| \\ &= \|(y_i - \bar{y}_i){\Sigma^i_{yy}}^{-\frac{1}{2}}{\Sigma^i_{yy}}^{\frac{1}{2}}\boldsymbol{\omega}\| \\ &\leq \|(y_i - \bar{y}_i){\Sigma^i_{yy}}^{-\frac{1}{2}}\|\|{\Sigma^i_{yy}}^{\frac{1}{2}}\boldsymbol{\omega}\| \\ &\leq \sqrt{\frac{m}{\tau}}\|{\Sigma^i_{yy}}^{\frac{1}{2}}\boldsymbol{\omega}\|\end{aligned} \tag{5.27}$$

it's obivious that the above geometric interpretation for the CTM constraint holds.

3. A similar geometric interpretation can be given for the SR constraint. Let

$$\widehat{\Sigma}^i_{yy} = \Sigma^i_{yy} + \mathbf{y}_i^T\mathbf{y}_i, \tag{5.28}$$

a SR constraint enforces each training data $\mathbf{y}_i$ picked from the ellipsoid $\Psi_i(0, \widehat{\Sigma}^i_{yy}, \sqrt{\frac{m}{\tau}})$

$$\begin{aligned}&\Psi_i(0, \widehat{\Sigma}^i_{yy}, \sqrt{\frac{m}{\tau}}) \\ \stackrel{\Delta}{=}&\{\mathbf{y}_i|\mathbf{y}_i(\widehat{\Sigma}^i_{yy})^{-1}\mathbf{y}_i^T \leq \frac{m}{\tau}\}\end{aligned} \tag{5.29}$$

satisfies the following inequality

$$\|\mathbf{e}_i\| \leq \xi_i + \varepsilon \tag{5.30}$$

This interpretation can be verified similarly to the CTM case:

$$\begin{aligned}\|\mathbf{e}_i\| &= \|y_i\boldsymbol{\omega}\| \\ &= \|y_i{\widehat{\Sigma}^i_{yy}}^{-\frac{1}{2}}{\widehat{\Sigma}^i_{yy}}^{\frac{1}{2}}\boldsymbol{\omega}\| \\ &\leq \|y_i{\widehat{\Sigma}^i_{yy}}^{-\frac{1}{2}}\|\|{\widehat{\Sigma}^i_{yy}}^{\frac{1}{2}}\boldsymbol{\omega}\| \\ &\leq \sqrt{\frac{m}{\tau}}\|{\widehat{\Sigma}^i_{yy}}^{\frac{1}{2}}\boldsymbol{\omega}\|\end{aligned} \tag{5.31}$$

From Theorem 5.2, we have

$$\begin{aligned}\left\| \begin{matrix} \left(\Sigma_{yy}^{i}\right)^{\frac{1}{2}}\boldsymbol{\omega} \\ \bar{\mathbf{h}}(\mathbf{x}_i)\boldsymbol{\beta} - \bar{\mathbf{t}}_i \end{matrix} \right\|^2 &= \|\Sigma_{ee}^{i} + \bar{\mathbf{e}}_i^T \bar{\mathbf{e}}_i\| \\ &= \|\boldsymbol{\omega}^T(\Sigma_{yy}^{i} + \mathbf{y}_i^T\mathbf{y}_i)\boldsymbol{\omega}\| \qquad (5.32) \\ &= \|\boldsymbol{\omega}^T(\Sigma_{yy}^{i} + \mathbf{y}_i^T\mathbf{y}_i)\boldsymbol{\omega}\| \\ &\leq \frac{\tau}{m}\left(\xi_i + \varepsilon\right)^2\end{aligned}$$

Taking square roots of (5.32) yields

$$\|\left(\widehat{\Sigma}_{yy}^{i}\right)^{\frac{1}{2}}\boldsymbol{\omega}\| \leq \sqrt{\frac{\tau}{m}}\left(\xi_i + \varepsilon\right) \qquad (5.33)$$

which together with (5.31) implies

$$\|\mathbf{e}_i\| \leq \xi_i + \varepsilon \qquad (5.34)$$

5.1.5 Robust ELM for Regression

Based on the preliminary results of last section, we now formulate CTM-constrained RELM (CTM-RELM) and SR-constrained RELM (SR-RELM) for noisy input and output data.

5.1.5.1 CTM based RELM

By adding second order moment constraints to the basic ELM formulation in Theorm 5.1, the CTM-RELM is formulated as

$$\begin{aligned} \min_{\boldsymbol{\beta},b,\theta,\boldsymbol{\xi}} \quad & L_P = b + C\sum_{i=1}^{N}\xi_i + D\sum_{i=1}^{N}\theta_i \\ s.t. \quad & \|\mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} - \mathbf{t}_i\| \le \varepsilon + \xi_i \\ & \|\left(\Sigma_{yy}^{i}\right)^{\frac{1}{2}}\boldsymbol{\omega}\| \le \theta_i\sqrt{\tau/m} \\ & \xi_i \ge 0 \qquad\qquad i = 1, 2, \ldots, N \\ & \|\boldsymbol{\beta}\| \le b \end{aligned} \tag{5.35}$$

where C is defined in (3.7), and D is a penalty coefficient to control the deviation of the prediction errors.

5.1.5.2 SR based RELM

Likewise, Theorem 5.2 also leads to a SOCP problem formulation:

$$\begin{aligned} \min_{\boldsymbol{\beta},b,\boldsymbol{\xi}} \quad & L_P = b + C\sum_{i=1}^{N}\xi_i \\ s.t. \quad & \left\| \begin{matrix} \left(\Sigma_{yy}^{i}\right)^{\frac{1}{2}}\boldsymbol{\omega} \\ \bar{\mathbf{h}}(\mathbf{x}_i)\boldsymbol{\beta} - \bar{\mathbf{t}}_i \end{matrix} \right\| \le (\xi_i + \varepsilon)\sqrt{\tau/m} \\ & \xi_i \ge 0 \qquad\qquad i = 1, 2, \ldots, N \\ & \|\boldsymbol{\beta}\| \le b \end{aligned} \tag{5.36}$$

5.1.6 Covariance in the feature space

We firstly calculate the covariance when the non-linear mapping functions are known explicitly. We write $\mathbf{h}(\mathbf{x})$ as follows

$$\mathbf{h}(\mathbf{x}) = [G(\mathbf{a}_1, b, \mathbf{x}), \ldots, G(\mathbf{a}_L, b, \mathbf{x})] \tag{5.37}$$

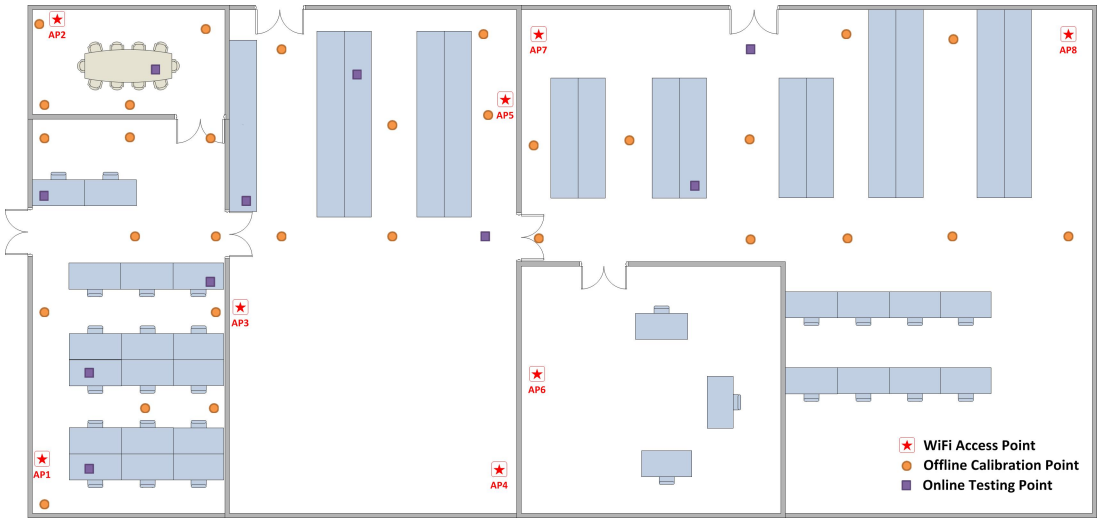

Figure 5.2: Positions of the WiFi access points, offline calibration points and online testing points in the test-bed

where $\mathbf{a}_i$, b are randomly generated weights and bias connecting an input and the ith hidden node. $G(\mathbf{a}_i, b, \mathbf{x})$ is the activation function.

A statistical method is provided to derive the covariance theoretically in the feature space. For each input $\mathbf{x}_i$ we randomly generate Z samples $\{\mathbf{x}_i^1, \mathbf{x}_i^2, \ldots, \mathbf{x}_i^Z\}$ according to the distribution of $\mathbf{x}_i$ with mean $\bar{\mathbf{x}}_i$ and covariance Σ_{xx}^i. Then the covariance matrix of $\mathbf{h}(\mathbf{x}_i)$ can be approximated by

$$\Sigma_{hh}^i = \frac{1}{Z}\sum_{z=1}^{Z} \tilde{\mathbf{h}}(\mathbf{x}_i^z)^T \tilde{\mathbf{h}}(\mathbf{x}_i^z) \tag{5.38}$$

where

$$\tilde{\mathbf{h}}(\mathbf{x}_i^z) = \mathbf{h}(\mathbf{x}_i^z) - \frac{1}{Z}\sum_{z=1}^{Z} \mathbf{h}(\mathbf{x}_i^z) \tag{5.39}$$

5.2 Performance Evaluation

The setup of the real-world IPS is similar to the one in Section 4.2.2. Four performance measures are introduced: mean root square error (MRSE), standard deviation (STD), worst case error (WCE) and repeatability (REP) over r repeated

realizations. Noted that MRSE , STD and WCE in this case are taken from the mean over the r repeated realizations. r in our experiment is selected as 30.

$$MRSE = \frac{1}{r}\sum_{j=1}^{r}\left(\frac{1}{s}\sum_{i=1}^{s}\left\|\mathbf{t}_i - \mathbf{h}_i\hat{\boldsymbol{\beta}}\right\|\right)_j$$

$$STD = \frac{1}{r}\sum_{j=1}^{r}\left(\sqrt{\sum_{i=1}^{s}\left(\left\|\mathbf{t}_i - \mathbf{h}_i\hat{\boldsymbol{\beta}}\right\| - \frac{1}{s}\sum_{i=1}^{s}\left\|\mathbf{t}_i - \mathbf{h}_i\hat{\boldsymbol{\beta}}\right\|\right)^2}\right)_j$$

$$WCE = \frac{1}{r}\sum_{j=1}^{r}\left(\max_{i\in S}\left\|\mathbf{t}_i - \mathbf{h}_i\hat{\boldsymbol{\beta}}\right\|\right)_j$$

$$REP = \sqrt{\frac{1}{r}\sum_{j=1}^{r}\left(\frac{1}{s}\sum_{i=1}^{s}\left\|\mathbf{t}_i - \mathbf{h}_i\hat{\boldsymbol{\beta}}\right\| - MRSE\right)_j^2}$$

where s is the number of testing samples, S is the index set of testing samples like $[1, 2, \cdots, s]$.

We apply our RELMs to the real-world data set, and compare our proposed algorithms with basic ELM, OPT-ELM and SVR [68]. In the CTM-RELM formulation, there are three hyperparameters, C, D and τ to be tuned. C and D are both selected by grid method from the exponential sequence $[2^{-5}, 2^{-4}, \cdots, 2^5]$ utilizing 5-fold cross-validation on the training data set. τ increases from 0.1 to 1 with a step size of 0.1. In SR-RELM case, there are two hyper-parameters, C and τ to be tuned, they are all selected with the same strategy as CTM-RELM. For both RELMs, the slack variable ε is empirically selected as 0.05. The SOCP problems are solved by CVX Matlab toolbox [60]. Since the performances of ELM and its variants are not sensitive to the number of hidden nodes L as long as it is larger than some threshold [69], we fix L as 500 for our proposed algorithms, basic ELM and OPT-ELM to facilitate the comparison of computational costs. The width of Gaussian kernel λ used in SVR is selected from the exponential sequence $[2^{-5}, 2^{-4}, \cdots, 2^5]$ utilizing 5-fold cross-validation.

The testing results with respect to four performance measures are shown in Table 5.1. Fig.5.3 illustrates the comparison in terms of cumulative percentile of error

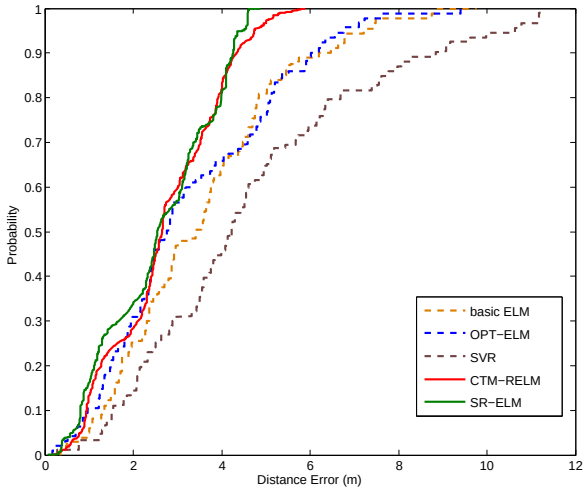

Figure 5.3: Cumulative percentile of error distance for IPS testing results

distance, which shows that the proposed CTM-RELM can provide higher accuracy and have an obvious effect in reducing the standard deviation compared to ELM and OPT-ELM. On the other hand, SR-RELM also gives an accuracy as good as CTM-RELM, and has better performance of confining the worst case error. The above results are reasonable, since the two robust constraints have their different emphasis. In addition, both CTM-RELM and SR-RELM can give better performances in REP than basic ELM. The enhancement of the repeatability is due to more constraints brought in the proposed algorithms, which shrinks the size of solution searching space.

The proposed algorithms incur longer training time due to the introduction of second order moment constraints instead of linear constraints. However, a slightly longer training time is not a concern in IPSs, considering that it is the calibration phase, e.g., procedure of radio map generating, that accounts for the large body of time consumption. Besides, RELMs inherit the simpleness, e.g., random feature mapping, dispensation with bias b, and single layer structure from ELM, therefore its training time is still competitive compared with SVR and its variants.

Table 5.1: Comparison of Experimental Testing Results

Performance measures	MRSE (m)	STD (m)	WCE (m)	REP (m)	Training Time (s)	Testing Time (s)
Basic ELM	3.71	3.84	8.22	0.79	1.13	7.1×10^{-3}
OPT-ELM	3.32	1.98	9.65	0.56	2.84	8.8×10^{-3}
SVR	4.66	2.64	10.93	0.024	4137.68	6.1×10^{-1}
CTM-RELM	2.96	1.26	5.61	0.12	141.65	9.7×10^{-3}
SR-RELM	2.94	1.53	4.47	0.22	18.88	9.0×10^{-3}

5.3 Discussion and Conclusions

Before concluding our work, we provide some important discussions.

1. Choice of the measure for accuracy: It's noteworthy that, we adopt MRSE instead of the conventional root mean square error (RMSE) as our measure. It's because MRSE makes more practical sense than RMSE for IPSs, which has been widely adopted in indoor positioning contests [70]. The measure of repeatability is introduced in particular for ELM because it produces variation in repeated realizations, namely, with same parameters setting, e.g., the number of hidden nodes, of the same training set, ELM may draw different results. This is mainly due to the reason that the number of hidden units is not infinite so that the universal approximation using SLFNs with random nodes may not be accurate [53]. However, it's should be noted that, most iteratively tuning based algorithms such as BP, actually also face the unreproducibility issue, and from the perspective of STD, ELM is even more stable.

2. Implementation tricks for RELMs: How to calculate the covariance and mean is tricky for regression problems, since one has to use only one sample to approximate its corresponding statistics. In this thesis, we take advantage of the specificity of the learning problem in IPSs - grouping. The whole data set can be divided into several groups by their belonging calibration points, and in any group, its members "theoretically" should have the same RSS (input) and

coordinates (output). But in reality, it's impossible due to the uncertainties as discussed above. However, these members in one certain group can be intuitively used to calculate the mean and covariance needed to represent the group for problem formulations. By this "grouping" trick, we can further reduce the number of the constraints in Equation (5.35) and Equation (5.36) from n to $\frac{N}{g}$, where g is the size of a group the number of sampling at one calibration point.

3. Inherent problem of ELM: ELM with given number of hidden nodes cannot approximate any complex functions or classify any complex planes in the feature space. Our proposed RELMs neither solve this problem in theory.

   Another inherent problem with ELM is its random selection of the input layer parameters which makes its output a random variable and may result in variations in performance for different realizations. Note that our proposed RELMs don't eradicate this problem in theory, but alleviate it by imposing SOCP constraints.

To sum up, we proposed CTM-RELM and SR-RELM to address the problem of noisy measurements in IPSs by introducing two CTM and SR constraints to the OPT-ELM, and further gave two SOCP based formulations. Simulation results and real-world indoor localization experiments both demonstrated that the CTM-RELM based IPS can provide higher accuracy and smaller standard deviation than other algorithms based IPSs; while the SR-RELM based IPS can provide better accuracy and smaller worst-case errors. The repeatability of the proposed algorithms was also demonstrated to be better.

Chapter 6

Development of a Zonal Occupancy Monitoring System

In *Chapters 4–5*, we focused on the indoor positioning problem under a stochastic framework to improve the accuracy and robustness to noises.

In this chapter, we further explore how to monitor the indoor occupancy using WiFi technology. We design a lightweight WiFi based monitoring system, which is free of users' cooperation. A novel clustering algorithm is employed to select cluster centers. Then, localization algorithms are leveraged to compute the locations of cluster centroids, which are also taken as the positions of clusters. Ultimately, the nearly real-time [1] positions and size (number of members) are grouped together and then served as the inputs of optimization and control of HVAC systems.

6.1 Preliminaries

Clustering refers to the process of partitioning a set of objects into subsets consisting of similar objects. The similarity measure between pairs of objects can be represented as either the cosine of the angle or Euclidean distance. Going into the details

[1] As the actuation of HVAC systems takes time, usually 1 to 2 minutes, so the monitoring systems need not to react in real-time. Nearly real-time occupancy monitoring systems are more suitable for HVAC systems

of all kinds of clustering algorithms is beyond the scope of this thesis, thus here we focus on the one recently proposed by Rodriguez et al. [71]. For convenience of description, we henceforth refer to it as FSFP, as the key insight of [71] is to perform clustering by the approach of Fast Searching and Finding of density Peaks.

The basic assumptions for FSFP are that cluster centers are surrounded by neighbors with lower local density and they are at a relatively large distance from any points with a higher local density, thus it belongs to the category of density-focused clustering algorithms. For each data point i, we calculate its local density ρ_i and its distance σ_i from points of higher density. Both these quantities depend only on the Euclidean distance d_{ij} between data points. The local density ρ_i is defined as

$$\rho_i = \sum_j \exp(-(d_{ij}/d_c)^2) \tag{6.1}$$

where d_c is a cutoff distance, i.e., a hyper-parameter specified by users. σ_i is measured by computing the minimum distance between the point i and any other point with higher density:

$$\sigma_i = \min_{j:\rho_j>\rho_i} (d_{i,j}) \tag{6.2}$$

For the point with highest density, we assign $\max_j(d_{ij})$ to its σ.

The core of FSFP is the observation that σ_i is much larger than the typical nearest neighbor distance only for points that are local or global maxima in the density. Thus, cluster centers are recognized as points for which the value of σ_i is anomalously large. Some clustering algorithms introduce a noise-signal cutoff to remove those low-density clusters [72,73]. By contrast, FSFP firstly determines a border region for each cluster, defined as the set of points assigned to that cluster but being within a distance d_c from data points belonging to other clusters. For each cluster, the point of highest density within its border region. We denote its density by ρ_b. Then the points of the cluster whose density is higher than ρ_b are considered part of the cluster core. The others are taken as part of the cluster halo, namely individual points that belong to nobody.

Compared with other popular clustering algorithms, the following reasons make FSFP more appropriate to be employed for our case:

1. A number of clustering algorithms, e.g. K-means [74] and K-medoids [75] methods, are notorious in that their performance are highly correlated with the selection of hyper-parameters, e.g., number of clusters K or density threshold. However, the only hyper-parameters involved in FSFP is the cutoff distance d_c, which implies in large data sets, the results of the analysis are robust w.r.t. the choice of d_c.

2. Due to the fact that FSFP uses the density ρ as the metric to determine clusters, it can capture any shapes of clusters while numbers of existing distance metric based approaches can capture spherical ones only.

3. FSFP outperforms iteration based clustering algorithms because it avoids local optima and lightweight in terms of computation.

6.2 Development of the Monitoring System

A novel feature of our monitoring system is that the existing commercial WiFi Access Points (APs) are upgraded to WiFi finders, which are able to detect the RSS packets transmitted between each mobile device and WiFi routers. Our monitoring system is independent of the connectivity between occupants' devices and certain WLAN networks, however it can still capture the RSSI and MAC address as long as the WiFi function of devices is enabled.

Our WiFi network consists of master AP and client APs (WiFi finders). Two critical functions are conducted by the master AP: one is for the for communications with client APs and additional servers;[2] the other one is when additional servers are not available, it can be responsible for calculation tasks, e.g., clustering and localization tasks in our case. It should be noted that, the naive mode of our monitoring system

[2]In our system, the server is optional and excluded by default.

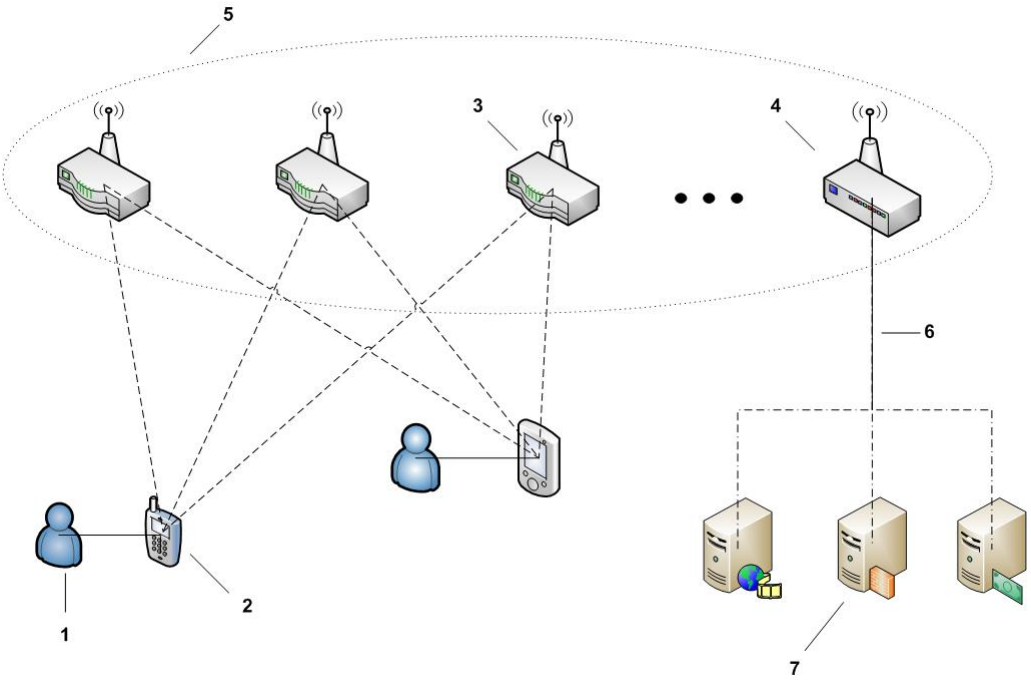

Figure 6.1: System Architecture of the proposed monitoring system: 1&2. Occupant and his WiFi-enabled mobile device; 3. WiFi client AP; 4. Master AP; 5. WiFi Sniffer Network; 6. Data transmission via UDP protocol (Optional); 7. Optional servers

is server-free, but it's compatible with additional servers when there are some specific needs, e.g., HVAC server for HVAC actuation based on the monitoring results.

Favored by the adopted dynamic (online) fingerprinting radio map, no calculation is performed on the server side. In contrary, the calibration schemes adopted by traditional fingerprinting algorithms usually require to store the heavy radio map, including the RSS packets and their corresponding physical positions of reference points. Instead, in our system, the RSSs of client APs are scanned along with other mobile devices, thus the timeliness of the reference radio information (radio map) is guaranteed. In addition, considering the number of client APs is much less than that of calibration points, the data required to be stored is the physical positions of client APs , which indicates that the ROM of a master AP is large enough for storing.

The system architecture of our WiFi based monitoring system is demonstrated in Figure 6.1.

Unlike most existing IPSs which the APs signal strengths are perceived by the mobile devices, for our system, the WiFi finders can capture a much larger number of RSS data (effectiveness can be achieved to millisecond level) as long as the mobile device is transmitting data. The firmware developed in the monitoring system is OpenWrt [76], thus It is compatible for most of chimerical brands of WiFi routers. All the data packets, RSS and the corresponding MAC address of each mobile device is analyzed according to the 802.11a protocol.

6.3 Device Filtering Algorithms

One ground assumption for WiFi based monitoring system is the bijection between mobiles devices and indoor occupants. Inaccurate predicted number of occupants can happen to our monitoring system due to invalidating this assumption. Nevertheless, it's common for an occupant to possess more than one WiFi enabled device and bring them to the target environment. Hence, before running the monitoring system, it's necessary to differentiate mobile devices, e.g., laptops, tablets and smartphones, at the first hand, then remove those devices which are not phones.

Generally, the indoor occupants can be grouped into two categories: long-term occupants, who have personal spaces in the target indoor environment, and short-term occupants, who are visitors to the target indoor environment. We assume that the short-term occupants don't bring mobile devices other than their mobile phones. Our device filtering algorithm is to remove those long-occupants' devices which are not phones, and henceforth, we refer to them as long-term disturbance devices (LDDs), and we further give the definition of activated access point as follows:

Definition 6.1. *Activated access point (AAP). We define an AP is an AAP by qth mobile device, or AAP_q for simplicity, if it has detected the qth mobile device at least half* [3] *of the days we run the device filtering task.*

[3]This parameter is case-by-case due to the different settings of indoor environmrnts

After running the filtering task for a week, we noted the fact that mobile phones tend to possess more AAPs than laptops or tablets. The reason behind this can be explained as that the mobile phones can migrate along with their occupants from one AP to another in different rooms, which may activate more APs in different places than other mobile devices. This is consistent with the example that the AP placed in the meeting room are hardly activated by WiFi enabled laptops or tablets. The side benefit brought by this filtering scheme is that it can also eliminate some vertical disturbances brought by those mobile phones of occupants in other floors, as WiFi signals attenuate dramatically when it penetrate through walls.

Algorithm 6.1: Pseudo-code of Device Filtering Algorithm

Input:
MD: Mobile devices detected during days running device filtering
D: Days of sensed, a vector with length N, where N is the number of MDs.
NAAPs: Total number of APs activated by devices during device filtering process, a vector with length N.
Output:
LDD: long-term disturbance devices
/*X_q *denotes the qth element of X in the following**/
Initialization:
$t \Leftarrow hyperparamter_1$: threshold of number of AAPs to determine devices
$TD \Leftarrow hyperparamter_2$: time (in day) of collecting data
Main program:
for Every MD **do**
if $D_q > TD/2$ **then**
if $NAAP_q < t\times$ Number of all APs **then**
$LDD \Leftarrow MD_q$ /*Add MD to the collection of LDDs*/
end if
end if
end for

6.4 Clustering Based Zonal Occupancy Detection Algorithm

When the device filtering process is completed, , we hardcode the MAC address of the found LDDs in our system to eliminate them, or simply create a blacklist

whose members will not be recorded when we run the monitoring system. Since the occupancy detection algorithm is always performed after device filtering, in this section, the RSS samples are referred to the RSS samples of mobile phones.

As discussed in the Section 6.2, the developed system is capable of finding RSSs of mobile devices in a passive-cooperation manner. Those RSS samples are inputs of the clustering algorithm. A problem of FSFP is that it can not automatically choose the cluster centers but involves the manual intervention. Although the authors of [71] pointed out that the quantity $\gamma_i = \rho_i \delta_i$ might be a good indicator, it doesn't perform well in the scenarios with sparse data points. Nevertheless, the indoor occupancy distribution in the commercial education buildings tends to be sparse compared to the area of the sites, which incurs the low value of ρ_i of cluster centers. By the fact that cluster centers have anomalously large value of δ, we modify the FSFP by sorting δ_i first then target the breakpoint br where the largest first-order difference happens, and the points whose δ is lager than δ_{br}, are selected as centers, which is in an automatic manner of clustering.

Once the RSS samples are received, FSFP [4] will give the results of clustering, including clusters' sizes and their respective centers.

The WKNN algorithm [33] is leveraged to calculate the positions of the cluster centers via the fresh radio map built on the dynamic RSSs of client APs. By the boundary information of zones, the centers of clusters can be further classified to their corresponding areas. The HVAC system can leverage the results of this zonal occupancy detection algorithm, e.g., which zones are occupied (active zones) and how many people are in the room (intensity) to perform energy efficient control.

[4] For terminology consistency, the FSFP adopted in the rest of this chapter is referred to the modified version.

6.5 Performance evaluation

6.5.1 Setup

We conducted real-world experiments to evaluate the performance of the proposed zonal occupancy monitoring system. The testbed is the Internet of Things Laboratory in the School of Electrical and Electronic Engineering, Nanyang Technological University. The area of the test-bed is around $580m^2$ ($35.1m \times 16.6m$). The scenario of the experiment, including placement of clients APs and places of occupants, is shown in Fig. 6.2.

Algorithm 6.2: Pseudo-code of Clustering Based Zonal Detection Algorithm

Input:
R: RSS feature matrix of size $S \times S$, where S indicates the number of client APs
T: RSS feature matrix of size $N \times S$, where N is the number of testing samples
Output:
ActZone: active zones
IntenZone: the intensity of activity of active zones
Initialization:
$d_c \Leftarrow hyperparamter_1$: cut-off distance of FSFP
$K \Leftarrow hyperparamter_2$: number of nearest neighbors used in WKNN
$Zone$: boundary information of zones
Pos_{ref}: the 2-dimensional Cartesian coordinates of reference points
Main program:
for Every m minutes **do**
(Centroids, SizeOfCluster) $\Leftarrow$ **function** FSFP(**T**; d_c)
CenPos $\Leftarrow$ **function** WKNN(Centroids, **R**, Pos_{ref}; K)
for $i = 1 : NumberOfClusters$ **do**
for $j = 1 : NumberOfZones$ **do**
if $CenPos_i \in Zone_j$ **then**
$ActZone \Leftarrow Zone_j$
IntenZone $\Leftarrow SizeOfCluster_i$
end if
end for
end for
end for

Based the locations of ventilation terminals and the existing boundaries in the environment, e.g., doors and walls, the partition of the testbed is provided in Fig. 6.3.

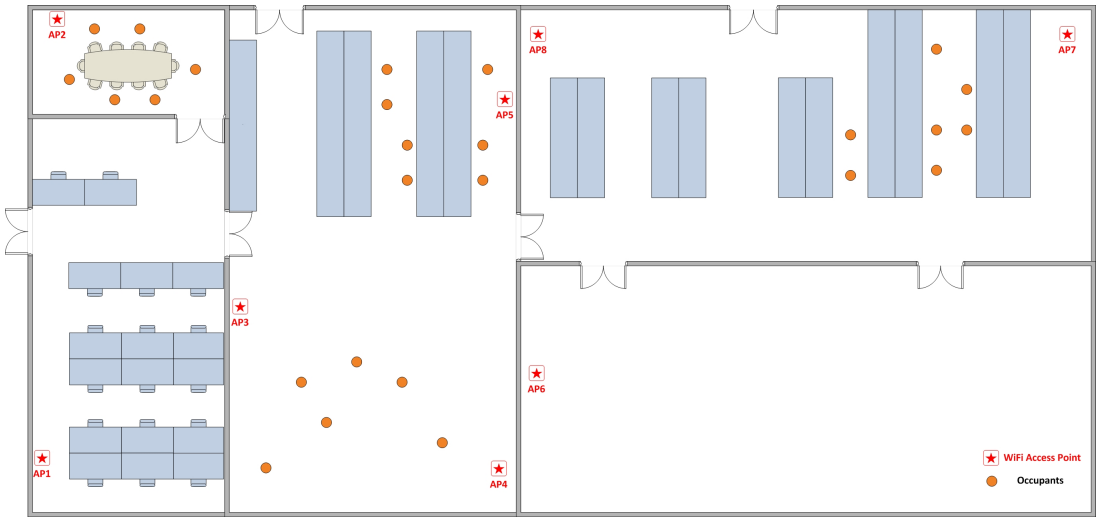

Figure 6.2: Scenario of Evaluation

The average area of zones is around $64m^2$, and we denote by the resolution of a partitioning as the square root of the average area of zones, which is $8m$ in this case.

6.5.2 Evaluation results

We evaluated the accuracy of our system by manually putting 26 mobile devices in 4 certain zones of Fig. 6.3. Since this evaluation was for accuracy of the devices whose MAC addresses were marked as testing samples, we omitted the procedure of device filtering.

Each mobile device represented a dummy occupant and they were put in 4 selected zones. The reference points in this case were 8 client APs and their RSS samples were online collected in each experiment. 30 experiments were conducted in the testbed though a five days evaluation period. To account for the signal variation caused by environmental changes, experiment for all 5 days were evaluated over the time horizon $10am, 12am, 2pm, 4pm, 6pm$ and $8pm$.

The experimental results in the 4 selected zones are illustrated as Fig. 6.4. Overall, our system can provide 92.5% accuracy. For the timestamps where they were off

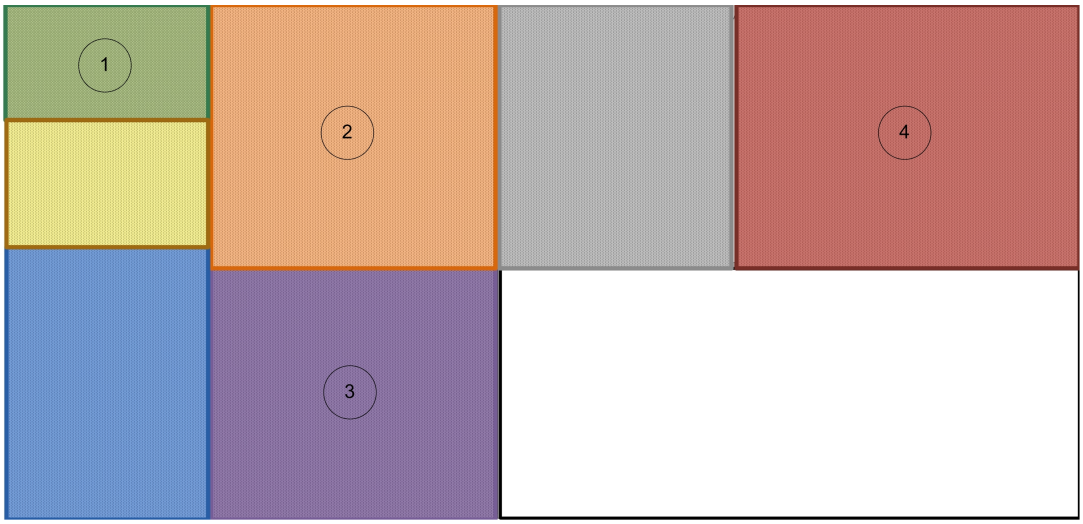

Figure 6.3: Partitioning of the testbed for evaluation, different colours correspond to different zones and the white space means blocked area

the ground truth, the errors were normally below 20% of the true size of the cluster, which is tolerable for most cases.

It's worthwhile to discuss the third experiment of $Zone2$, where we got a prediction that 0 occupant was in the $Zone2$. It implies that the coordinate predicted by WKNN of the cluster center of $Zone2$ was wrong, and according to the boundary information, it's further assigned to a wrong zone. We observe that this sort of mistake happens when the ground truth of the selected cluster centroids are close to boundaries of some zones.

By comparing the $6th$ and $29th$ timestamps of $Zone2$ and $Zone3$, we found that the reason for the errors of two zones simultaneously since one point belonging to $Zone3$ was mis-clustered to $Zone2$. Similarly, this also happens to those points close to boundaries. How to handle the points close to boundaries will be studied in the future work.

Regarding the computational complexity, as we take the master AP as our "server", the time of the calculation is within 1.5 seconds, implying it's qualified for any real-time needs of occupancy monitoring.

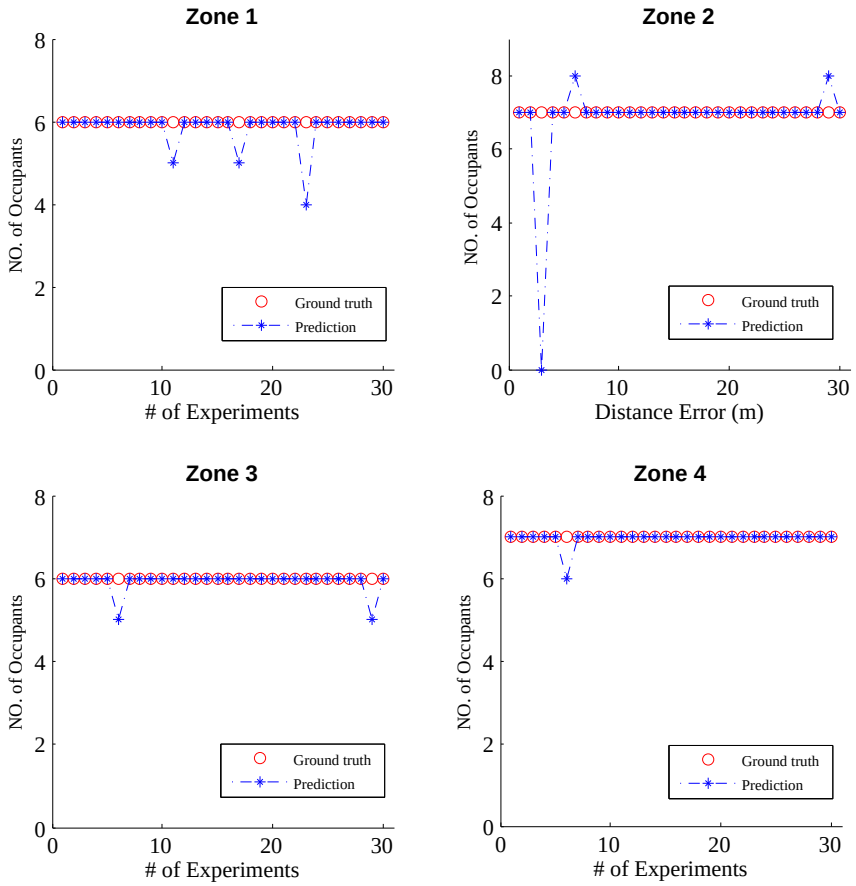

Figure 6.4: Comparison of prediction and ground truth of EV1

6.6 Conclusions

In this chapter, we designed a zonal occupancy monitoring system based on clustering algorithm by existing WiFi infrastructures. Our system is computationally lightweight, because the role of central processor in the system is played by the master AP, which is only a commercial router. The information of client APs, including physical positions and their online RSS samples, is used as radio map, which makes our system robust to environmental dynamics.

To make the bijection between occupants and mobile devices valid, disturbed WiFi enabled devices, e.g. tablets and laptops, should be removed. We proposed a device filtering algorithm, which utilizes the relationship between activated APs and different types of mobile devices to make this filtering possible.

The results of multi-time experiments demonstrate that our system can give accurate and reliable predictions with fast speed.

In the future, we will test our system in higer-resolution testbed in larger areas. Meanwhile, how to alleviate the problem of signal variation and misclassification of centroids close to boundaries will be further studied.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

The explosive proliferation of mobile devices and the popularity of social networks have spurred extensive demands on Location Based Services (LBSs) in recent decades. Our team has developed an indoor positioning system by leveraging WiFi techniques. Three indoor localization algorithms, OS-ELM, DZ-ELM, RELM have been proposed and implemented into real-world indoor positioning systems. Beyond IPSs, we also designed a zonal occupancy monitoring system for building energy efficiency, which is lightweight and free of calibration. The main results are summarized as follows:

1. DZ-ELM was introduced in Chapter 4 to address uncertainties such as ambient dynamics. Meanwhile, replacing the quadratic function with a function containing dead zone can relieve the effects of outliers caused by signal variation more adaptive to different types of noises. Experimental and simulation results in IPSs demonstrated that DZ-ELM could provide higher accuracy and better repeatability than ELM.

2. In Chapter 5, we studied the uncertainties in both inputs and outputs. The uncertainties are inevitable in IPSs due to many reasons, e.g., manual observational errors, signal variation and ambient dynamics. RELMs were proposed from a more general viewpoint to enhance the robustness of indoor localization algorithms. We developed two different types of RELMs respectively, i.e. CTM-RELM and SR-RELM, based on different robust constraints. The simulation results and real-world indoor localization experiments both demonstrate that CTM-RELM based IPS can provide higher accuracy and smaller standard deviation than the basic ELM based and OPT-ELM based ones; while SR-ELM based IPS can provide better accuracy and smaller worst-case errors. The repeatability of the proposed algorithms was also demonstrated to be better.

3. The knowledge of occupancy levels in discrete zones within a building offers the potential of significant energy savings when coupled with zonal control of building air-conditioning and lighting services, thus we designed a clustering based zonal occupancy monitoring system by existing WiFi infrastructure in Chapter 6. Our system is lightweight in computation and run on a TP-Link router. In addition, as we adopted client APs as reference points, our radio map is dynamic and robust to environmental changes. To make the bijection between occupants and mobile devices valid, a device filtering algorithm was also proposed to remove those devices which are not mobile phones. The system was run in a testbed with resolution of $8m$, experimental results demonstrate that our system is accurate and fast that can be further incorporated to various real-time energy control systems.

7.2 Future Work

We list some possible directions for future studies.

- Fingerprinting approach is time-consuming and involves labour intensive surveying of sites to generate radio maps. In Chapter 6, we successfully leveraged the client APs as reference points to make our system calibration-free, however, this method can not be applied to a positioning system as the density of client APs may be low and most positioning problems are regression based. Essentially, the propagation model of radio is non-linear, while WKNN still falls into the scope of linear approximation. A systematic way, from the perspective of non-linear expansion, needs to be further explored to make the calibration-free possible for IPSs.

- The scalability problem of IPS is never trivial. Since it's impossible to utilize all the routers in a large-scale scenarios due to communication and computation problems. Future work will be devoted to finding a sparse way, based on submodularity, to efficiently make use of APs for positioning and monitoring.

Author's Publications

Journal papers:

1. **X. Lu**, H. Zou, H. Zhou, L. Xie, G. Huang, "Robust Extreme Learning Machine for Indoor Localization Systems", accepted in *IEEE Transactions on Cybernetics*, 2015.

2. H. Zou, **X. Lu**, H. Jiang, and L. Xie, "A Fast and Precise Indoor Localization Algorithm based on Online Sequential Extreme Learning Machine", *Sensors*, 15, 1804-1824, 2015.

3. H. Zou, B. Huang, **X. Lu**, H. Jiang and L. Xie, "A Robust Indoor Positioning System based on the Procrustes Analysis and Weighted Extreme Learning Machine", submitted to *IEEE Transactions on Wireless Communication*, 2014.

Conference papers:

1. **X. Lu**, C. Yu, H. Zou, H. Jiang, L. Xie, "Extreme Learning Machine with Dead Zone and its Application to WiFi Based Indoor Positioning", in 13*th IEEE International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2014.

2. **X. Lu**, Y. Long, H. Zou, C. Yu, L. Xie, "Robust Extreme Learning Machine for Regression Problems with Noisy Measurements in WiFi based Indoor Positioning System", in 24*th IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2014.

3. H. Zou, H. Jiang, **X. Lu**, and L. Xie, "An Online Sequential Extreme Learning Machine Approach to WiFi Based Indoor Positioning", in *IEEE World Forum on the Internet of Things (WF-IoT)*, 2014.

Bibliography

[1] LAUNCH, "Google Maps Takes a Stab at Bing with Indoor Maps Feature for Airports and Malls," http://blog.launch.co/blog/google-maps-takes-a-stab-at-bing-with-indoor-maps-feature-fo.html.

[2] R. Mautz, "Indoor positioning technologies," Ph.D. dissertation, Habil. ETH Zürich, 2012, 2012.

[3] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems Man and Cybernetics Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1067–1080, 2007.

[4] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 13–32, 2009.

[5] G. Deak, K. Curran, and J. Condell, "A survey of active and passive indoor localisation systems," *Computer Communications*, vol. 35, no. 16, pp. 1939–1954, 2012.

[6] H. Zou, L. Xie, Q.-S. Jia, and H. Wang, "Platform and algorithm development for a rfid-based indoor positioning system," *Unmanned Systems*, vol. 2, no. 03, pp. 279–291, 2014.

[7] H. Zou, H. Jiang, X. Lu, and L. Xie, "An online sequential extreme learning machine approach to wifi based indoor positioning," in *IEEE World Forum*

on Internet of Things (WF-IoT). IEEE, 2014, Conference Proceedings, pp. 111–116.

[8] Department of Energy (DoE), "Energy efficiency trends in residential and commercial building," http://apps1.eere.energy.gov/buildings/publications/pdfs/corporate/bt_stateindustry.pdf.

[9] Nationa Environment Agency (NEA), "Energy Efficiency in Singapore," http://www.nea.gov.sg/cms/ccird/E2%20Singapore%20(for%20upload).pdf.

[10] V. L. Erickson, M. Á. Carreira-Perpiñán, and A. E. Cerpa, "Occupancy modeling and prediction for building energy management," *ACM Transactions on Sensor Networks (TOSN)*, vol. 10, no. 3, p. 42, 2014.

[11] Y. Agarwal, B. Balaji, S. Dutta, R. K. Gupta, and T. Weng, "Duty-cycling buildings aggressively: The next frontier in HVAC control," in *10th International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 2011, Conference Proceedings, pp. 246–257.

[12] V. L. Erickson, M. . Carreira-Perpin, and A. E. Cerpa, "OBSERVE: Occupancy-based system for efficient reduction of HVAC energy," in *10th International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 2011, Conference Proceedings, pp. 258–269.

[13] X. Lu, Y. Long, H. Zou, C. Yu, and L. Xie, "Robust extreme learning machine for regression problems with its application to WiFi based indoor positioning system," in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2014, Conference Proceedings.

[14] T. Teixeira, G. Dublon, and A. Savvides, "A survey of human-sensing: Methods for detecting presence, count, location, track, and identity," *ACM Computing Surveys*, vol. 5, 2010.

[15] A. Bekkali, H. Sanson, and M. Matsumoto, "Rfid indoor positioning based on probabilistic rfid map and kalman filtering," in *Wireless and Mobile Com-*

puting, Networking and Communications, 2007. WiMOB 2007. Third IEEE International Conference on. IEEE, 2007, pp. 21–21.

[16] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 2. IEEE, Conference Proceedings, pp. 775–784.

[17] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, "Landmarc: indoor location sensing using active rfid," *Wireless networks*, vol. 10, no. 6, pp. 701–710, 2004.

[18] A. Bekkali and M. Matsumoto, "Rfid indoor tracking system based on inter-tags distance measurements," in *Wireless Technology.* Springer, 2009, pp. 41–62.

[19] A. Kushki, K. N. Plataniotis, and A. N. Venetsanopoulos, "Intelligent dynamic radio tracking in indoor wireless local area networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 3, pp. 405–419, 2010.

[20] Ekahau, *http://www.ekahau.com/.*

[21] A. Kushki, K. N. Plataniotis, and A. N. Venetsanopoulos, *WLAN Positioning Systems: Principles and Applications in Location-based Services.* Cambridge University Press, 2012.

[22] X. Li, K. Pahlavan, M. Latva-aho, and M. Ylianttila, "Comparison of indoor geolocation methods in dsss and ofdm wireless lan systems," in *Vehicular Technology Conference, 2000. IEEE-VTS Fall VTC 2000. 52nd*, vol. 6. IEEE, 2000, pp. 3015–3020.

[23] B. B. Peterson, C. Kmiecik, R. Hartnett, P. M. Thompson, J. Mendoza, and H. Nguyen, "Spread spectrum indoor geolocation," *Navigation*, vol. 45, no. 2, pp. 97–102, 1998.

[24] P. Stoica and R. L. Moses, *Introduction to spectral analysis.* Prentice hall Upper Saddle River, NJ, 1997, vol. 1.

[25] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *Proceedings of the 3rd international conference on Mobile systems, applications, and services*. ACM, Conference Proceedings, pp. 205–218.

[26] A. Varshavsky, E. de Lara, J. Hightower, A. LaMarca, and V. Otsason, "Gsm indoor localization," *Pervasive and Mobile Computing*, vol. 3, no. 6, pp. 698–720, 2007.

[27] J. Chung, M. Donahoe, C. Schmandt, I.-J. Kim, P. Razavai, and M. Wiseman, "Indoor location sensing using geo-magnetism," in *Proceedings of the 9th international conference on Mobile systems, applications, and services*. ACM, Conference Proceedings, pp. 141–154.

[28] A. Matic, A. Papliatseyeu, V. Osmani, and O. Mayora-Ibarra, "Tuning to your position: Fm radio based indoor localization with spontaneous recalibration," in *IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 2010, Conference Proceedings, pp. 153–161.

[29] H. Zou, L. Xie, Q.-S. Jia, and H. Wang, "An integrative weighted path loss and extreme learning machine approach to rfid based indoor positioning," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2013, Conference Proceedings, pp. 1–5.

[30] T. Roos, P. Myllymäki, H. Tirri, P. Misikangas, and J. Sievänen, "A probabilistic approach to wlan user location estimation," *International Journal of Wireless Information Networks*, vol. 9, no. 3, pp. 155–164, 2002.

[31] X. Chai and Q. Yang, "Reducing the calibration effort for probabilistic indoor location estimation," *IEEE Transactions on Mobile Computing*, vol. 6, no. 6, pp. 649–662, 2007.

[32] B. Ferris, D. Fox, and N. Lawrence, "WiFi-SLAM Using Gaussian Process Latent Variable Models," in *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, 2007, pp. 2480–2485.

[33] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[34] T. Roos, P. Myllymaki, and H. Tirri, "A statistical modeling approach to location estimation," *IEEE Transactions on Mobile Computing*, vol. 1, no. 1, pp. 59–69, 2002.

[35] A. Goswami, L. E. Ortiz, and S. R. Das, "Wigem: a learning-based approach for indoor localization," in *Proceedings of the Seventh Conference on emerging Networking EXperiments and Technologies*. ACM, 2011, p. 3.

[36] C. K. Seow and S. Y. Tan, "Non-line-of-sight localization in multipath environments," *IEEE Transactions on Mobile Computing*, vol. 7, no. 5, pp. 647–660, 2008.

[37] Y. Gwon, R. Jain, and T. Kawahara, "Robust indoor location estimation of stationary and mobile users," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 2. IEEE, 2004, pp. 1032–1043.

[38] H. Wen, Z. Xiao, N. Trigoni, and P. Blunsom, "On assessing the accuracy of positioning systems in indoor environments," in *Wireless Sensor Networks*. Springer, 2013, pp. 1–17.

[39] O. Woodman and R. Harle, "Pedestrian localisation for indoor environments," in *Proceedings of the 10th international conference on Ubiquitous computing*. ACM, 2008, pp. 114–123.

[40] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: zero-effort crowdsourcing for indoor localization," in *Proceedings of the 18th annual international conference on Mobile computing and networking*. ACM, 2012, pp. 293–304.

[41] F. Evennou and F. Marx, "Advanced integration of wifi and inertial navigation systems for indoor mobile positioning," *Eurasip journal on applied signal processing*, vol. 2006, pp. 164–164, 2006.

[42] Z. Xiao, H. Wen, A. Markham, and N. Trigoni, "Lightweight map matching for indoor localization using conditional random fields," in *The International Conference on Information Processing in Sensor Networks (IPSN'14)*. ACM, 2014.

[43] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: unsupervised indoor localization," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 2012, pp. 197–210.

[44] G. Shen, Z. Chen, P. Zhang, T. Moscibroda, and Y. Zhang, "Walkie-markie: indoor pathway mapping made easy," in *Proceedings of the 10th USENIX conference on Networked Systems Design and Implementation*. USENIX Association, 2013, pp. 85–98.

[45] Y. Agarwal, B. Balaji, S. Dutta, R. K. Gupta, and T. Weng, "Duty-cycling buildings aggressively: The next frontier in hvac control," in *10th International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 2011, pp. 246–257.

[46] K. Weekly, N. Bekiaris-Liberis, and A. M. Bayen, "Modeling and estimation of the humans' effect on the CO_2 dynamics inside a conference room," *arXiv preprint arXiv:1403.5085*, 2014.

[47] W. J. Fisk, "A pilot study of the accuracy of CO_2 sensors in commercial buildings," *Lawrence Berkeley National Laboratory*, 2008.

[48] B. Balaji, J. Xu, A. Nwokafor, R. Gupta, and Y. Agarwal, "Sentinel: occupancy based HVAC actuation using existing WiFi infrastructure within commercial buildings," in *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*. ACM, Conference Proceedings, p. 17.

[49] S. Goyal, H. A. Ingley, and P. Barooah, "Occupancy-based zone-climate control for energy-efficient buildings: Complexity vs. performance," *Applied Energy*, vol. 106, pp. 209–221, 2013.

[50] J. Chen and C. Ahn, "Assessing occupants energy load variation through existing wireless network infrastructure in commercial and educational buildings," *Energy and Buildings*, vol. 82, pp. 540–549, 2014.

[51] G.-B. Huang, "An insight into extreme learning machines: random neurons, random features and kernels," *Cognitive Computation*, pp. 1–15, 2014.

[52] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 13, pp. 489–501, 2006.

[53] G.-B. Huang, L. Chen, and C.-K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Transactions on Neural Networks*, vol. 17, no. 4, pp. 879–892, 2006.

[54] M.-B. Li, G.-B. Huang, P. Saratchandran, and N. Sundararajan, "Fully complex extreme learning machine," *Neurocomputing*, vol. 68, pp. 306–314, 2005.

[55] G. Huang, S. Song, J. N. Gupta, and C. Wu, "Semi-supervised and unsupervised extreme learning machines," *IEEE Transactions on Cybernetics*, 2014.

[56] G.-B. Huang, X. Ding, and H. Zhou, "Optimization method based extreme learning machine for classification," *Neurocomputing*, vol. 74, no. 1, pp. 155–163, 2010.

[57] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[58] A. J. Smola and B. Schlkopf, "A tutorial on support vector regression," *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.

[59] R. Lozano-Leal and R. Ortega, "Reformulation of the parameter identification problem for systems with bounded disturbances," *Automatica*, vol. 23, pp. 247–251, 1987.

[60] M. C. Grant, S. P. Boyd, and Y. Ye, "Cvx: Matlab software for disciplined convex programming (web page and software)," *Available at http://cvxr.com/cvx*.

[61] K. Forsman and L. Ljung, *The Dead Zone in System Identification*. Springer US, 1996, book section 5, pp. 69–82.

[62] G. Durgin, T. S. Rappaport, and H. Xu, "Measurements and models for radio path loss and penetration loss in and around homes and trees at 5.85 ghz," *IEEE Transactions on Communications*, vol. 46, no. 11, pp. 1484–1496, 1998.

[63] N. Kothari, B. Kannan, E. D. Glasgwow, and M. B. Dias, "Robust indoor localization on a commercial smart phone," *Procedia Computer Science*, vol. 10, pp. 1114–1120, 2012.

[64] W. Meng, W. Xiao, W. Ni, and L. Xie, "Secure and robust wi-fi fingerprinting indoor localization," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, Conference Proceedings, pp. 1–7.

[65] P. K. Shivaswamy, C. Bhattacharyya, and A. J. Smola, "Second order cone programming approaches for handling missing and uncertain data," *The Journal of Machine Learning Research*, vol. 7, pp. 1283–1314, 2006.

[66] H. Gao, S. Shiji, W. Cheng, and Y. Keyou, "Robust support vector regression for uncertain input and output data," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 23, no. 11, pp. 1690–1700, 2012.

[67] J. Navarro, "A simple proof for the multivariate Chebyshev inequality," *arXiv preprint arXiv:1305.5646*, 2013.

[68] J. A. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle, J. Suykens, and T. Van Gestel, *Least squares support vector machines*. World Scientific, 2002, vol. 4.

[69] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Transactions on Systems Man and Cybernetics Part B: Cybernetics*, vol. 42, no. 2, pp. 513–529, 2012.

[70] Q. Yang, S. J. Pan, and V. W. Zheng, "Estimating location using Wi-Fi," *IEEE Intelligent Systems*, vol. 23, no. 1, pp. 8–13, 2008.

[71] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[72] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, *Automatic subspace clustering of high dimensional data for data mining applications.* ACM, 1998, vol. 27, no. 2.

[73] L. Ertöz, M. Steinbach, and V. Kumar, *Finding Clusters of Different Sizes, Shapes, and Densities in Noisy, High Dimensional Data*, ch. 5, pp. 47–58.

[74] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Applied statistics*, pp. 100–108, 1979.

[75] L. Kaufman and P. J. Rousseeuw, *Finding groups in data: an introduction to cluster analysis.* John Wiley & Sons, 2009, vol. 344.

[76] "Openwrt," https://openwrt.org.