LEARNING HUMAN CONTEXTS THROUGH UNOBTRUSIVE METHODS

BY

CHENREN XU

A dissertation submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Doctor of Philosophy

Graduate Program in Electrical and Computer Engineering

Written under the direction of

Yanyong Zhang

And approved by

_____

_____

_____

_____

New Brunswick, New Jersey

October, 2014

© 2014

Chenren Xu

ALL RIGHTS RESERVED

ABSTRACT OF THE DISSERTATION

Learning Human Contexts through Unobtrusive Methods

By CHENREN XU

Dissertation Director:

Yanyong Zhang

Learning human contexts is critical to the development of many applications, ranging from healthcare, business, to social sciences. Most existing work, however, acquires contextual information in an obtrusive manner – they may require subjects to carry mobile devices, or rely on self or peer report to report data. In this dissertation, we present two unobtrusive techniques that can help us learn important human contextual information including count, location, trajectory, and speech characteristics. We first present SCPL, a radio frequency-based device-free localization technique. SCPL is able to count how many people are in an indoor setting and track their locations by observing how they disturb the wireless radio links in the environment. Second, we present Crowd++, a smartphone-based speech sensing technique, which records a conversation and automatically counts the number of people in the conversation without prior knowledge of their speech characteristics. Both techniques are unobtrusive, low-cost, and private, which can thus enable a large array of important applications that rely upon the knowledge of human contextual information.

Acknowledgements

During the past years of my Ph.D. journey, there are so many people I would like thank for their help. First of all, I would like to thank my advisor Yanyong Zhang for her support, the time she spent to advice, guide, and teach me how to seek ambitious goals and accomplish academic achievements. She always tried her best to provide me with a relaxed and balanced research environment to achieve academic freedom. She made a great impact and influence on my philosophy, and her guidance magically reshapes me into a highly self-motivated and independent scholar pursuing a academic career in the future. My Ph.D. journey won't be so enlightened, enjoyful and fruitful without her.

Richard.E.Howard added significant great value to my work with his "universal" knowledge, especially on physics, wireless and hardware. I gained enormous benefits from his years' experience from Bell Labs about experimental science in the very beginning stage of my Ph.D. He always encourages me to try my best as much as I can to understand the fundamentals when encountering any problem. I would like to thank him for being so generous to me with his time and expertise.

A large portion of my work initializes from the collaboration between WINLAB and industry labs. I would like to thank all the industry researchers including Jun Li from Technicolor Labs, and Emiliano Miluzzo and Yih-Farn Chen from AT&T Labs. Their guidance and insights greatly broaden my view in conducting high-impact research towards real-world applications.

I would like to thank Wade Trappe, Richard Martin and Yingying Chen for their time serving on my committee.

I would also like to thank Wade Trappe for generous support for my last year Ph.D. I also wish to thank many collaborators, colleagues and friends, who have influenced

my views and efforts during my time inside and outside Rutgers, including Giovanni Vannucci, Marco Gruteser, Xiaodong Lin, Richard Martin, Ning An, Bernhard Firner, Robert Moore, Sugang Li, Gang Liu, Ashwin Ashok, Mingchen Gao, Jing Lei, Tam Vu, Feixiong Zhang, Zhuo Chen, Dan Zhang, Prashant Jadhav and Kai Su.

I would like to express my earnest gratitude to my parents and my family members, all of whom made countless sacrifices to raise me and to give me the best possible education. They have always been a rock behind me, encouraging, loving, and supporting me unconditionally. This thesis is dedicated to them.

Finally, I owe immense gratitude to my wife, for her endurance to put up with my often hectic life style during my Ph.D. I do not think I could have made it without her. Her presence and strength gave me the drive to accomplish this. Thank you for everything.

Dedication

To my wife Jinwei Wu and my parents Zhenguang Xu and Qianfen Hu, for their love, continuous support and encouragement.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

1.1.1 The Quest for "Unobtrusive" Human Context Sensing

Learning human context information is vital to our daily life as it covers almost all our daily aspects, such as where I am, what I am doing, and who I am with. Looking back, the term context-aware computing was first introduced by Schilit [61] following Mark Weiser's vision of ubiquitous computing [69] over two decades ago. This vision foresees an intelligent world that computers would become such an integral part of our environment that we won't be aware of them anymore. The recent decade has witnessed the trend where smart sensors are increasingly embedded in sensing infrastructure (e.g., thermostat, motion sensor, smoke detector) and mobile devices (e.g., phones, tablets, glass, watches). As a result, this vision is becoming tangible. Smart sensors at homes or commercial buildings provide isolated but focused sensing functionalities, such as temperature, room occupancy and smoke. On the other side, the ones on our mobile devices are seamlessly making our life easier by means of supporting our mobility: with a suite of sensors (accelerometer, microphone, GPS, WiFi, camera, digital compass, gyroscope), the device can provide built-in or third-party services targeting at a variety of contexts, such as user's physical location (latitude and longitude), contextual location (home and work place, indoor and outdoor etc.), relative orientation and location between a device and user, etc. These technologies can lead to revolutions in different domains, such as healthcare, entertainment, transportation, and social networks.

However, we argue that despite the progress, there is still much room to improve

in this area. Specifically, we would like to minimize the inconveniences cased by these technologies. For example, existing solutions often incur the following issues:

- *Need to wear a device:* Many current techniques rely on the user to carry a device all the time. For example, location services assume the user will always be co-located with the device, which is not the case when users are at home. Smart technology should be able to continuously track people wherever they are.
- *Extensive calibration:* In the area of human-centric computing, the characteristics of each individual is not completely uniform and not necessarily known by the system beforehand. Profiling the system with all the possible scenarios and will be a straightforward solution, but it usually requires extensive calibration and frequency recalibration to be sustainable to the environmental change.
- *Need to report status manually:* Social science, such as the fields of psychology and human computer interaction [51], involves the study, planning, design and uses of the interaction between people and computers. For example, to study children's autism, researchers often record audio/video during study. Afterwards, they manual listen to or watch the recordings to by get groundtruth, which process is both labor intensive and not necessarily accurate.

As a result, the dissertation statement is that we would like to develop and evaluate "unobtrusive" techniques to learn human contexts such that the vision of "context-aware computing" is brought close to realization.

1.1.2 Proposed Solutions

In this dissertation, we propose two techniques to unobtrusively sense human contexts such as physical activities and social activities including such as count, location, trajectories, speed, and speaker count.

Radio-Frequency (RF) Based Device-free People Counting and Localization

Passive Localization In Cluttered Environments Using Classification Methods [78]

In today's world, regardless where we are, we are constantly exposed to RF signals emitted from a variety of radio sources. The presence of human subjects (without wearing any device) can interfere with these signals in such a way that by observing how the received signal strength (RSS) changes over time, we can infer people's locations. Since no device is required on the subject, we call this localization method RF-based device-free localization.

Passive localization is challenging, especially in cluttered indoor environments (home, office, etc.), because radio is radiating omni-directionally, and the receiver will thus receive signals not only from the visible Line-of-Sight (LoS) component but also from invisible multipaths (as a result of reflection, diffraction, and/or scattering). As a result, even a small move by the subject may lead to a great RSS fluctuation at the receiver, making it challenging to infer locations from the observed RSS values. We address this challenge by shifting the localization problem to a cell-identification problem – we partition the area into cells based on contextual information such as a cubicle, sofa, or bed, and then identify in which cell is the subject by solving a probabilistic classification problem. By choosing an appropriate cell size, we can balance the localization accuracy with calibration overhead. We tested our algorithm in a one-bedroom apartment and an office, achieving a sub-meter localization accuracy for a single person by having a device every 5 m^2.

Counting and Localizing Multiple People [76]

Device-free localization can localize people when they move about in their daily life without specifically wearing a device for localization purposes. In these scenarios, it is important to be able to localize multiple people at the same time. More importantly, this should be done without the need to calibrate multiple subjects, which will lead to a factorial growth in the calibration effort. Our objective is then to localize multiple people using a single subject's calibration data.

We achieve this objective by first counting how many people are in the room, and then calculating their locations. We propose a successive cancellation based scheme to count people. We first detect whether there is more than one subject in the room

by looking at the RSS values; If yes, we assume there is only one person and use the probabilistic method described above to identify his/her cell number. Then we subtract this subject's impact on the RSS values from the total RSS change. We repeat this process until there is no subject left in the remaining RSS change. Once we know the number of people in the room, we model indoor human trajectories as a state transition process, exploit indoor human mobility constraints from the site map, and integrate all information into a conditional random field (CRF) to calculate their locations at the same time. We show that our algorithm achieves an 86% of average counting accuracy and about 1 meter localization accuracy for tracking up to four people.

Unsupervised Speaker Counting on Smartphones [80]

Smartphones can serve as an excellent mobile social sensing platform, with the microphone in particular being exercised in several supervised audio inference applications, such as speaker identification, emotion detection, etc. Recognizing the fact that the most direct form of social interaction occurs through the spoken language and conversations, we identify it is important to automatically estimate how many people are involved in a conversation. Speaker count specifies the number of people that participate in a conversation, which is one of the primary metrics to evaluate a social setting: how crowded is a restaurant, how interactive is a lecture, or how socially active is a person. Thus, we developed Crowd++, an audio inference service running on off-the-shelf smartphones to accurately extract the speaker count from recorded audio data, without any supervision, and in different use cases. Through extensive experiments in real settings, we demonstrate that Crowd++ can to accurately estimate the number of people talking in a certain place with an average error distance of 1.5 speakers.

1.2 Contribution

Herein, we make several contributions to the unobtrusive human context sensing field, as summarized in the following:

- We designed and conducted extensive field experiments to study how indoor

radio multi-path effect brings challenges in device-free human localization problems, and understood the limitations of applying geometric family approach in this problem. We further designed *PC-DfP*, a probabilistic algorithm framework formulating this localization problem into a supervised machine learning classification problem. PC-DfP reduces the human calibration effort and mitigate the error caused by the multi-path effect by discretizing the physical space into context-based cells and taking random samples from different locations to average out the deep fading effect. PC-DfP further maintains high localization accuracy in long-term deployments by identifying and selecting the radio links robust to the environmental change.

- To enable this RF-based device-free localization technique to scale, we further designed SCPL, a technique simultaneously count and localize multiple people, which is unique in at least four contributions: (i) to our knowledge, it is the first work to systematically perform simultaneous counting and localization for up to four device-free people (moving or stationary) in large-scale deployments only using RF-based techniques; (ii) we designed a set of algorithms to count and localize multiple subjects relying on the calibration data collected by only a single individual; (iii) We also use plausible trajectory constraints (e.g. not walking through walls) based on floor map information, and integrate this information into the radio calibration data to further improve the tracking accuracies; and (iv) we recognize the nonlinear fading effects caused by multiple subjects in cluttered indoor environments, and design the algorithms to mitigate the resulting error.

- We oversaw the potentials of the "speaker count" context information in a number of social sensing applications, designed and prototyped a mobile application called *Crowd++* running on off-the-shelf smartphones. Crowd++ is unique given its number of contributions: (i) it is entirely distributed, with no infrastructure support; (ii) it applies completely unsupervised learning techniques and no prior training is needed for the system to operate; (iii) it is self-contained, in that, the sensing and machine learning computation takes place entirely and efficiently

on the smartphone itself as shown by our implementation on four different Android smartphones and two tablet computers; (iv) it is accurate, as shown by experiments where Crowd++ is used in challenging environments with different audio characteristics - from quiet to noisy and loud - with the phone both inside and outside a pocket, and very short audio recordings; and (v) it's energy and resource-efficient.

1.3 Organization

The rest of this dissertation is organized as follows. In Chapter 2, we present a measurement study of the impact of radio multipath propagation on the received signal strength when human are exposed to a radio environment. Then we describe the PC-DfP algorithm in more detail and share our experiences how it works in realistic indoor environments. Next, Chapter 3 first presents a measurement study of how multiple people collectively interfere with the received signal strength in radio environments, and impresents how SCPL algorithm leverages the calibration data collected with one person and the room map information to count and localize multiple people. Finally, Chapter 4 describes our speaker counting technique in more detail and share our lessons learned from the large scale experimental results from real world scenarios.

Chapter 2

PC-DfP: A Probabilistic Classification Approach for Device-free Human Localization

The proposed method for localzing one person using radio signal strength, namely PC-DfP, is presented in this chapter. This method will play an fundamental role for counting and localizing multiple people discussed in next chapter as well.

2.1 Introduction

There is growing interest in incorporating automatic "intelligence" in our homes and offices using a dense array of wireless radio/sensor nodes. Central to this intelligence is often the need to localize and track people in indoor environments. Many radio frequency (RF) based localization techniques have been proposed, such as those discussed in [5, 74, 17, 30, 59, 63, 81, 39, 52, 13, 82, 47, 68]. Most of these techniques, however, require the subjects to carry wireless devices, and are referred to as device-based active localization. This requirement has several inherent disadvantages. First, tracking stops whenever the device is detached from the subject either accidentally or intentionally. Second, for applications such as elder care, we cannot assume the subjects will always agree or remember to carry the device.

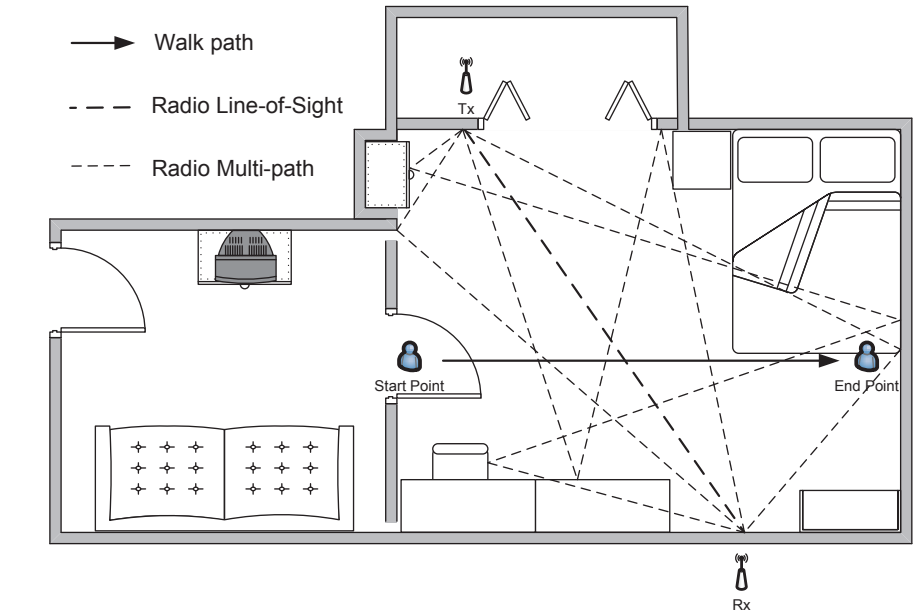
Recognizing these limitations, the community has started the discussion on RF-based device-free passive (DfP) localization techniques [83]. Compared to its active localization counterpart, DfP offers a lower cost solution as it does not require the participation of the subject and uses low-power RF devices that may already be available in our home/office environment. In DfP localization, we capture the change of the RF signals caused by the subject and try to derive his/her location based upon this

change.

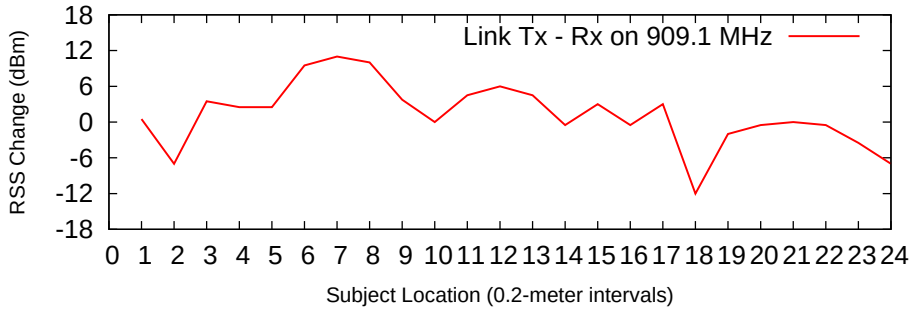
Deriving a subject's location from the RSS change caused by the subject, however, is a challenging task, mainly due to the well-known "multi-path" effect [56] that is caused by the reflection and diffraction of the RF signal from subjects and objects in the environment.

Let us look at a simple experiment to understand the effect of the multipath problem. Figure 2.1(a) shows the topology of a one-bedroom apartment in which we conduct our experiments. We have one transmitter (Tx in the picture) and one receiver (Rx in the picture), and this radio link has one Line-of-Sight (LoS) component and four Non-Line-of-Sight (NLoS) (or, multipath) components. We only show four NLoS components for simplicity; in reality there are many more present. A person walks from the marked "Start Point" to the marked "End Point". During the movement, we record the received signal strength (RSS) at the receiver (operating at 909.1 MHz), and report the differences between these values and the RSS values when the subject is absent in Figure 2.1(b). Figure 2.1(b) shows that the person's effect on the RSS value is random and unpredictable – we observe RSS decreases at different levels, and sometimes we even observe an RSS increase. Figure 2.1(b) also shows that changes from motion relative to the LoS and NLoS components can be far larger when the subject is not on the LoS than when he is – the variation is as high as 10 db from location 17 to location 18 over a distance of less than 20 cm where the person is not crossing the LoS of the link. Finally, we note that the multipath effect is affected by many factors. Figure 2.1(c) shows a completely different behavior when the radio frequency is set to 433.1 MHz.

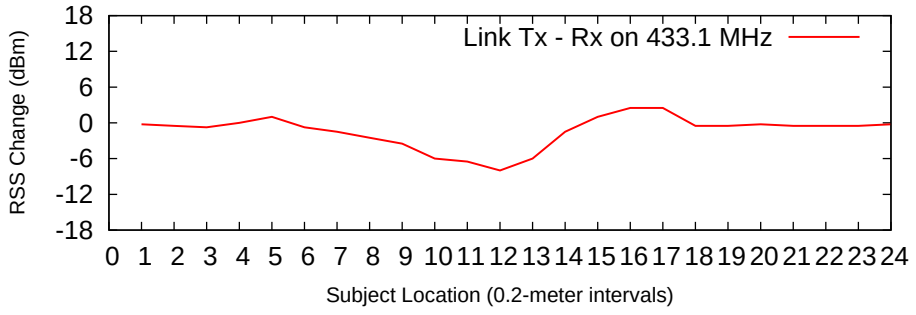
Many earlier DfP localization techniques either ignored multipath, or failed to treat multipath carefully enough. For example, radio tomography proposed in [71] tries to calculate a subject's location based upon the signal attenuation when the subject is blocking the LoS of the link. These schemes assume there is a direct relationship between a subject's location and the impact on radio signals. They will have good localization results either outdoor or in an empty room with little multipath. In a cluttered room, which is more common in real life than empty rooms, this assumption does not hold. In [83, 62], the authors acknowledge the importance of multipath, and propose



(a)



(b)



(c)

Figure 2.1: (a) shows the indoor environment in which the radio link has one LoS and four NLoS components; (b) and (c) show the fluctuation of RSS changes between Tx and Rx when the radios operate at 909.1 MHz and 433.1 MHz respectively.

a fingerprinting-based approach in which they first collect a radio map with the subject present in a few predetermined locations, and then map the test location to one of these trained locations based upon observed radio signals. While the fingerprinting approach is certainly a better fit for indoor DfP localization, the localization algorithm in [83, 62] adopts a point-based simplistic minimum Euclidean distance based matching algorithm, which is only practical when the training locations are sparse and the test location closely matches one of the training locations. As training points become denser, classification difficulty will grow significantly.

In this chapter, we take on the challenge and strive to improve the performance of DfP localization. Considering the complexity of multipath, *we choose to adopt the fingerprinting approach, and try to achieve good results when we have dense training locations, and random test locations.* We believe these requirements are crucial to many smart home applications such as infant care or elder care. We achieve improved results with the following two optimizations. First, we apply discriminant analysis to the classification problem based on the assumption that the covariates follow a multivariate Gaussian distribution. We validate the assumption of Gaussian distribution through experimental data as well as theoretical approximations. Second, in collecting radio signal readings, we adopt various ways to mitigate the multipath effect so that signal variations within a short distance become smoother. This can increase the distance between classes and further lead to higher classification likelihood. Specifically, our study has the following contributions:

- We derive a sophisticated classification model to better describe the DfP localization problem.
- We improve the quality of data sets by mitigating the error caused by the multipath effect.
- We show that in a one-bedroom apartment of 5×8 m that consists of 32 cells (each being 0.75×0.75 m in size), with 8 transmitters and 8 receivers, we can estimate the occupied cell ID with an accuracy of 97.2%.

- We show that our approach can achieve cell estimation accuracies over 90% in degraded conditions, such as reducing the training overhead (taking 16 data samples per cell instead of 100 samples), reducing the computation overhead, using fewer radio devices (10 devices instead of 16), and conducting tests a month later after the training.
- We show that our approach can be used to track multiple people when they are standing still, walking, sitting, or even lying down. We can also localize multiple people that co-exist in the apartment.
- We also implement our approach in a much larger commercial office space, and report a cell estimation accuracy of 93.8% from 32 cubicle-size cells.

The rest of the chapter is organized as follows. In Section 2.2, we highlight the challenges faced in indoor DfP localization. We model the system and present our localization algorithm in Section 2.3. In Section 2.4, we introduce our experimental setup and methodology. In Section 4.5, we implement our algorithm in a one-bedroom apartment and report detailed experimental results. We discuss the related studies in Section 4.7, and conclude the chapter in Section 4.8.

2.2 Challenges In a Cluttered Indoor Environment

In this section, through experimentation, we demonstrate the differences between RF-based outdoor and indoor localization, and highlight the challenges posed by indoor environments.

2.2.1 Outdoor Free Space Localization

We begin our experiments in an open outdoor environment. By setting up a transmitter and a receiver attached on tripods 4.5 meters away from each other in an empty parking lot, we only have a relatively small reflection from the ground. We partition the area into $0.75 \times 0.75\ m$ cells and categorize the cells into two groups: those on the LoS, and those off the LoS. We first record the median of the RSS measurements when

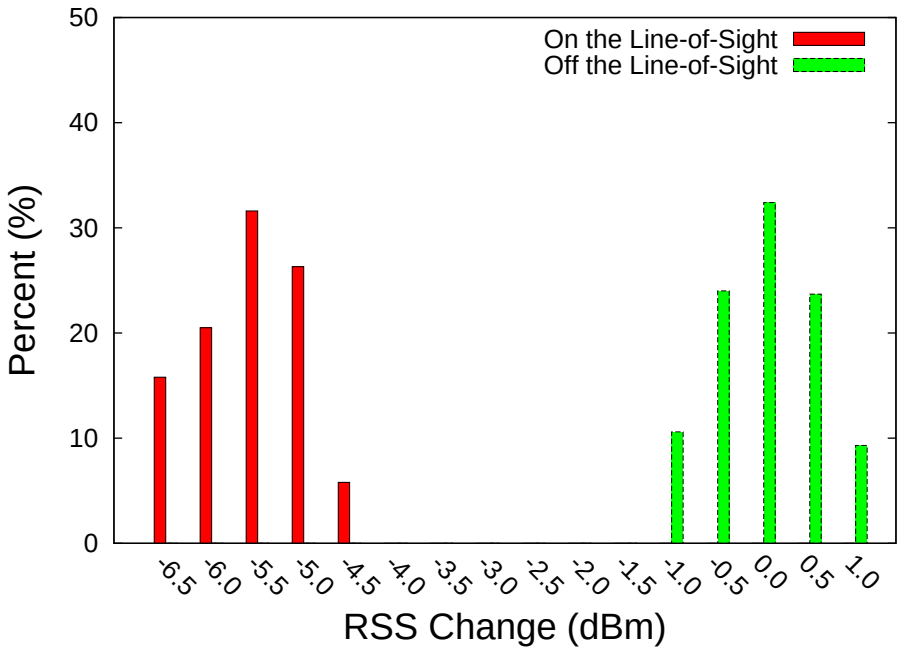


Figure 2.2: In an outdoor environment, when the radio devices are placed lower than the subject height, the subject causes distinctly different RSS changes for on-LoS cells and off-LoS cells.

the subject is 9 meters away from either device, RSS_E, which represent the base RSS when the subject is absent. Then, we collect 10 continuous RSS readings from each cell while the subject remains stationary in that cell. For each cell, we calculate the RSS change caused by the subject.

We first place the radios such that their height from ground is less than a person's height. In this way, a person can block radio signals more pronouncedly. Figure 2.2 shows that in this setting, RSS changes in different cells caused by the person clearly fall into two disjoint sets. RSS changes in on-LoS cells are much larger than RSS changes in off-LoS cells. This observation suggests that we may perfectly determine whether the subject is on the LoS or not simply by setting an appropriate threshold for observed RSS changes, which agrees with the observations in earlier studies [71, 11].

Next, we repeat the same experiment, but place the radios above the height of the subject (radios were placed 2 meters above the ground, and the subject is 1.8 meters in height). In this case, the position of the subject has little effect on the RSS values, as shown in Figure 2.3. As a result, in the rest of the study, we place the radio devices

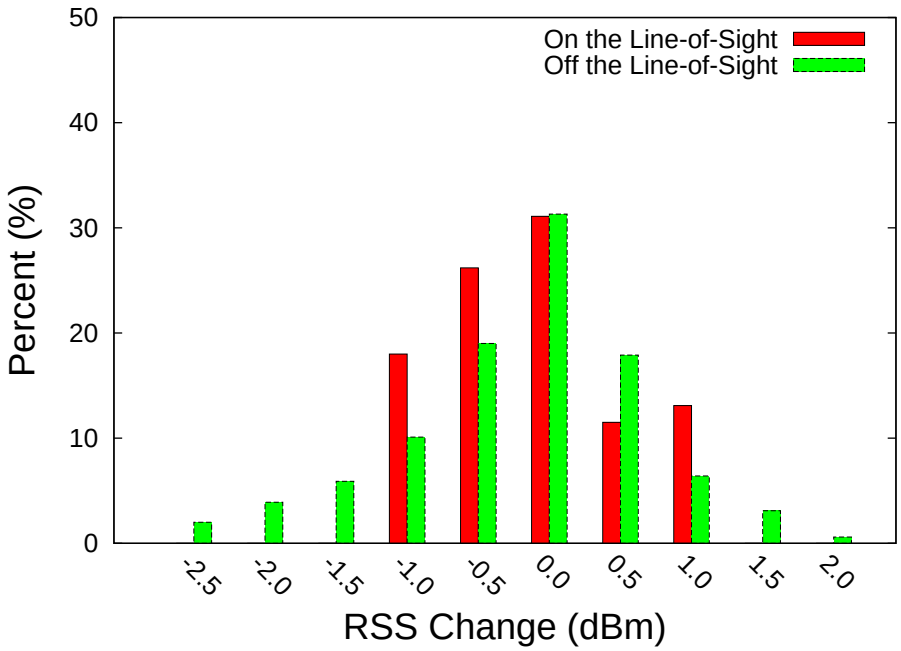


Figure 2.3: In an outdoor environment, when the radio devices are placed higher than the subject height, the subject causes little effect on the radio signals regardless of his location.

vertically lower than the subject except when explicitly noted.

2.2.2 The Multipath Effect

Compared with straightforward localization in the outdoor space, localization in the indoor space is much harder because of the multipath problem. This is particularly true for environments of interest for most applications. Next we will support this statement using experimental observations.

In our indoor experiments, we attach the transmitters and receivers on the wall, 1.2 meters above the floor, which is below most adults and above most of the furniture so that the impact of a subject's presence on the radio signal is maximized. As explained earlier, in an indoor environment, the subject may have an unpredictable impact on the RSS. Figure 2.4 shows the histogram of RSS changes in different cells. We observe that, when a subject randomly blocks a LoS, *there is only a 50% probability of the signal being attenuated by 1 dB or more*. In other words, 50% of the time the signal will not attenuate or even increase. This observation clearly shows that *the assumption of "blocking*

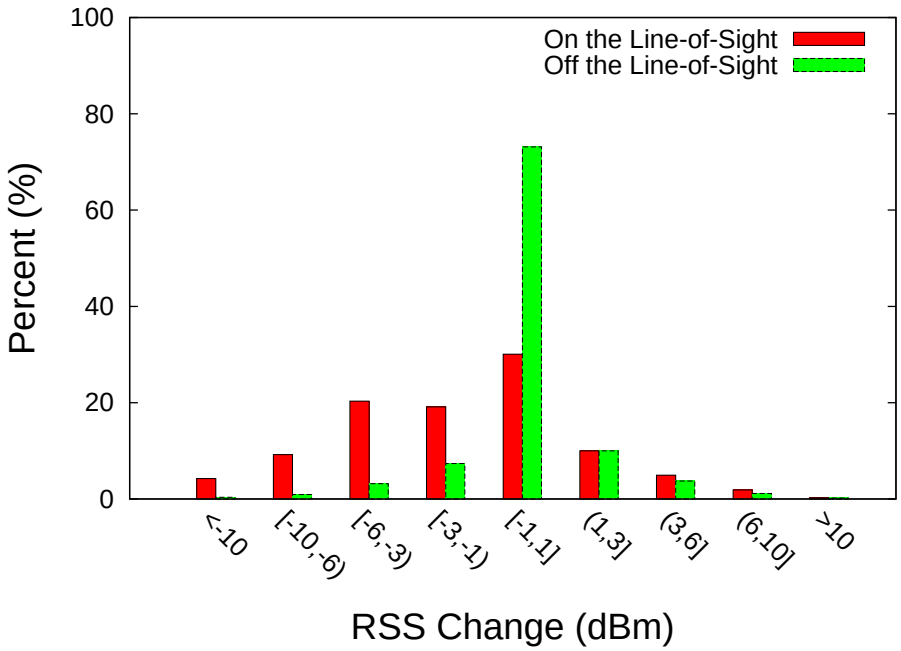


Figure 2.4: In an indoor environment, when the radio devices are placed below the subject height, the subject's effect on the radio signal is unpredictable with respect to his location.

LoS" means "attenuation" is misleading in cluttered environments. On the other hand, the results show that if a subject does not block any LoS, there is a 15% probability that the RSS of a radio link will change more than 3 dB. This further shows the unpredictable nature of the multipath effect.

2.3 Device-free Passive Localization through Probabilistic Classification Methods (PC-DfP)

As discussed earlier, indoor radio propagation is a very complex phenomenon such that the relationship between a subject's location and the resulting RSS of any radio links in the environment is hard to predict. Thus, statistical rather than deterministic methods are required to extract location information from the measured RF signals. In this section, we discuss in detail our probabilistic classification based device-free passive localization method, *PC-DfP* in short.

2.3.1 Overview of PC-DfP

We visualize a room as a grid of small square cells with unique addresses or ID numbers. By localizing a subject, we mean to estimate accurately the ID of the cell in which the subject is located. In our method, we assume there are L radio links in the environment, and there are K cells in a room. In the training phase, we first measure the RSS values for all L radio links when the room is empty (referred to as environmental RSS). Then for each cell k, we collect a set of RSS values with the subject present in this cell. The change between the environmental RSS and the RSS when the subject is in cell k, $[x_{k,1}, ..., x_{k,L}]$, gives the RSS change vector, $\mathbf{x_k}$, for cell k. $\mathbf{x_k}$ is referred to as the *footprint* for cell k. By the end of the training phase, we have obtained RSS footprints for every cell in the room. We build a K-class classifier based on the RSS footprints. Subsequently, in the testing phase, this classifier is used to classify the testing subject with an unknown label (i.e., cell ID).

2.3.2 Discriminant Analysis

In formulating our classification problem, we label a class k as the state with the subject in the k-th cell, with the associated RSS footprint $\mathbf{x_k}$. For each cell k, we collect the RSS footprint matrix X_k of dimension $\mathbb{R}^{n_k \times L}$, where n_k denotes the number of RSS footprints sampled in the training phase for the k-th cell. The class label is denoted as y_k. The goal of our analysis is to classify the subject with an unknown label into the correct cell ID based on the measured RSS vector.

A large number of classification techniques have been proposed in the literature, including density based approaches. Under the $0-1$ loss, the objective is to find the maximizer of the class posterior distribution $P(Y|X)$, where Y is the class label y_k and X is the RSS change vector $\mathbf{x_k}$. A simple application of the Bayes rule gives

$$P(Y = k|X = x) = \frac{f_k(x)\pi_k}{\sum_{j=1}^{K} f_j(x)\pi_j},$$

where $f_k(x)$ is the class-conditional density of X in class $Y = k$, and π_k is the prior

probability of class k that sums to 1. Assuming f to be a multivariate Gaussian distribution, we have the classical discriminant analysis. In the remaining of this section, we present a few variations of this technique and describe the rationale for applying them to solve our localization problem.

Minimum Euclidean Distance (MED)

Suppose we have the mean vector $\mu_k \in \mathbb{R}^L$ of the RSS for each class k from the training data. We also have the testing RSS vector x and $\hat{y}$ associated with the unknown cell label to be estimated. The Euclidean distance between x and μ_k is defined as

$$d(x, \mu_k) = \sqrt{\sum_{i=1}^{l} (x_i - \mu_{k_i})^2},$$

where

$$\mu_k = \sum_{i \in class\ k} x_i / n_k.$$

Thus, we have the objective classifier function

$$\hat{y} = argmin_k d(x, \mu_k),$$

as studied in [62].

Linear Discriminant Analysis (LDA)

Linear discriminant analysis aims to find a linear combination of features which characterize or separate two or more classes of subjects [21]. We assume the density of each class k is multivariate Gaussian with mean μ_k and a common covariance matrix Σ:

$$f_k(x) = \frac{1}{(2\pi)^{\frac{L}{2}} |\Sigma|^{\frac{1}{2}}} exp\left[-\frac{1}{2}(x - \mu_k)^T \Sigma^{-1} (x - \mu_k)\right].$$

Applying Bayes rule, we have the objective function

$$\hat{y} = argmax_k f_k(x) \pi_k.$$

In the log-scale, we can write the discriminant function as

$$\delta_k(x) = x^T\Sigma^{-1}\mu_k - \frac{1}{2}{\mu_k}^T\Sigma^{-1}\mu_k + \log \pi_k,$$

and we find

$$\hat{y} = argmax_k \delta_k(x).$$

Maximization of the discriminant function results in the following parameter updates:

- $\hat{\pi}_k = n_k/n$;
- $\hat{\mu}_k = \sum_{i \in class\ k} x_i/n_k$;
- $\hat{\Sigma} = \sum_{k=1}^{K} \sum_{i \in class\ k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T/(n-K)$;

In our experiment, the number of samples n_k is the same across the all the cells. Therefore the class probability $\pi_j = 1/K$ for all the classes.

Quadratic Discriminant Analysis (QDA)

In practice, it is rare that multiple classes share a common covariance matrix. Quadratic Discriminant Analysis (QDA) is a generalization of LDA that allows different covariance matrices. Such a generalization results in more flexible quadratic decision boundaries comparing to the linear decision boundaries from LDA. The resulting discriminant function is

$$\delta_k(x) = -\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \log \pi_k.$$

The flexibility of QDA comes with the cost of estimating the different covariance matrices Σ_k. When the dimensionality of x is high, this amounts to a huge increase on the number of parameters to be estimated. Thus in practice, with limited sample size, the simpler LDA is preferable.

Dimension Reduction

In practice, parameter estimation can be challenging even for LDA when data dimension is high. One way to address this problem is through feature selection or dimension reduction. Herein we adopt the linear projection scheme so that the L dimensional vector x can be projected to a q dimensional space via $z = Wx$, where W is a $q \times L$ matrix and $q < L$. For a fixed q, the optimal W is computed by maximizing

$$J(W) = \frac{W^T S_B W}{W^T S_W W},$$

where the within class scatter matrix is

$$S_W = \sum_k (\mu_k - \bar{\mu})(\mu_k - \bar{\mu})^T,$$

and the between class scatter matrix is

$$S_B = \sum_k \sum_{i \in class\ k} (x_i - \mu_k)(x_i - \mu_k)^T.$$

Here $\bar{\mu}$ is the overall mean of x, and μ_k is the mean of the kth class. This leads to solving an eigenvalue problem whose solution is $W_l = S_B^{-1/2} v_l$, where v_l is the lth eigenvector of $S_B^{1/2} S_W^{-1} S_B^{1/2}$. The resulting z is a compact representation of x in a lower dimensional space by projecting the original data to the *first q principal discriminant components*. In this way, we can minimize the localization error, reduce the computational cost and prevent the potential over-fitting and singularity problem.

2.3.3 Gaussian Approximation

In LDA and QDA, we assume that the conditional density given the class label is multivariate Gaussian. In this section, we first present experimental data to support this assumption, and then provide theoretical discussions on why our problem can be approximated by the Gaussian distribution.

Figures 2.5(a)-(c) show representative histograms for those links with RSS stable

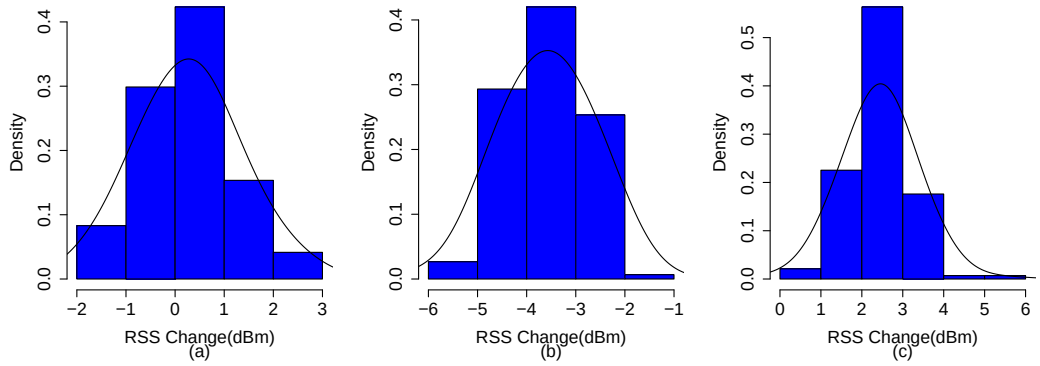


Figure 2.5: Three histograms for typical experimental RSS change measurements from an arbitrary link when a subject moves randomly within an arbitrary cell. The smooth curve is a log-normal density distribution.

(a), attenuated (b) and increased (c). We observe that most of the links fit the log-normal distribution well enough to produce an acceptable fit. As a result, treating RSS values (in power) as Gaussian is a valid assumption. The fact that our results based upon this assumption achieve good classification accuracies (as high as 97% shown in Section 4.5) is a further support for this assumption.

Next, we explain why we expect that a Gaussian model approximation would work as a first approximation in our classification problem. First, we note that the problem we are addressing is not a typical problem discussed in the literature [56], where the statistics of the multi-path signals at the receiver are considered when either the transmitter or receiver are moved, like in active RF-based localization problems. In passive localization, all the path lengths remain fixed, but the presence of a subject introduces attenuation, scattering, or diffraction of a subset of the multi-path signals. Based upon the geometry of the experimentation room and some simple measurements, we can make analogies, though, to the more typical multipath problem.

In Figure 2.7(b), it is clear that the major fraction of the links between transmitters and receivers have a substantially clear LoS or at most are obstructed by one relatively transparent interior partition wall. Because of the dominance of the large planar and often perpendicular reflecting surfaces (floor, walls, ceiling), one would expect the multi-path signals to be dominated by LoS and a few, relatively strong components,

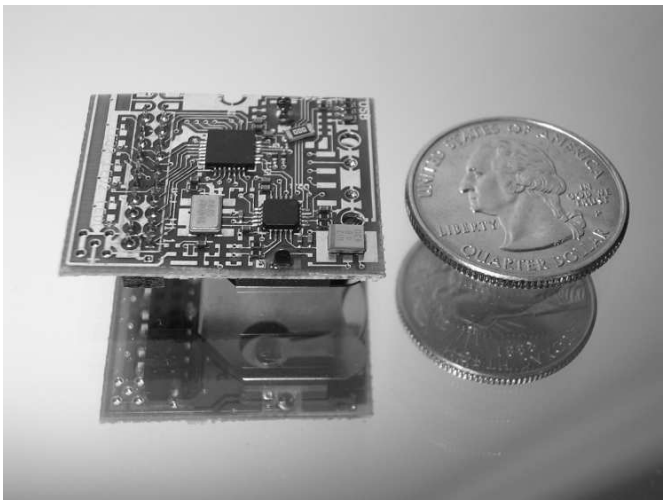
as seen in [8], along with many components so much smaller than the LoS component that they are insignificant. Finally, we note that in moving around, even a subject that is completely out of the LoS can cause a change in RSS of 10-20 db. This is consistent with a situation in which there are only a small number of multipath components of a magnitude large enough that they could add up constructively and cancel the LoS component to within 10% in amplitude, resulting in a 20db change in energy.

Extending a simplified Rician model [56] to our model would result a dominant LoS signal and a limited number of important multipath signals whose energy was somewhat smaller in total. This would be the Rician limit where the statistics of the signal are approximately Gaussian, as we have seen. Our results show that this approximation is adequate for our environments.

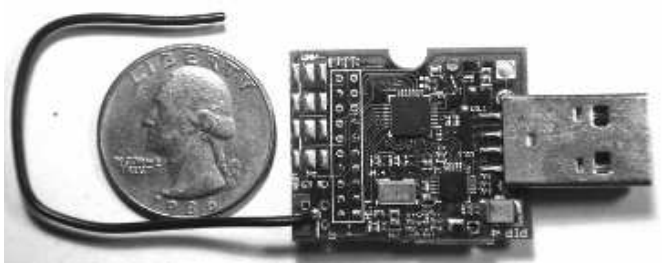
2.4 Experimental Methodology

In our experiments we will show that one or more subjects can be successfully localized in a home/office environment using our PC-DfP method. The system was deployed in two environments: a one bedroom apartment with home furniture and a commercial office space with cubicles and offices. Since most of the experiments were conducted in the first setting, we will focus on the first setting (i.e., the one-bedroom apartment) in the rest of the chapter unless otherwise noted. The apartment pictures are shown in Figures 2.7(a). The apartment is below ground level with a floor area of 5×8 m and a height of 3 m. The floor is concrete, the walls are wallboard on wooden studs, and the ceiling is acoustic tile.

Our experimental setup consists of a host PC (Intel i7-640LM 2.13GHz, 3GB RAM) serving as the centralized system, and eight transmitters and eight receivers. Receivers are connected to the PC through a (wireless) USB hub. In our system, each transmitter broadcasts a 10-byte packet every 100 milliseconds. The receivers will forward received packets to the host PC for data collection and analysis. In Section 4.5, we show that we can reduce the number of radio devices while maintaining good localization results.



(a)



(b)

Figure 2.6: (a) Wireless transmitter. (b) Wireless receiver with USB.

2.4.1 Hardware Description

The radio devices used in our experiments contain a Chipcon CC1100 radio transceiver and a 16-bit Silicon Laboratories C8051-F321 microprocessor powered by a 20 mm diameter lithium coin cell battery, the CR2032. The receivers have a USB connector for loss-free data collection but are otherwise identical to the transmitters. In our experiments, the radio operates in the unlicensed bands at 433.1 MHz or 909.1 MHz. Transmitters use MSK modulation, a 250kbps data rate, and a programmed output power of 0dBm. Each transmitter periodically broadcasts a 10-byte packet (8 bytes of sync and preamble and 2 bytes of payload consisting of transmitter's id and sequence number) ten times per second. When the receiver receives a packet, it measures the RSS values and wraps the transmitter id, receiver id, RSS, timestamp (on the receiver side) into a "data packet". This packet is sent to the centralized system over direct USB connection

or through network hub for data analysis. The transmitter and receiver are shown in Figure 2.6.

2.4.2 Experimental Setup

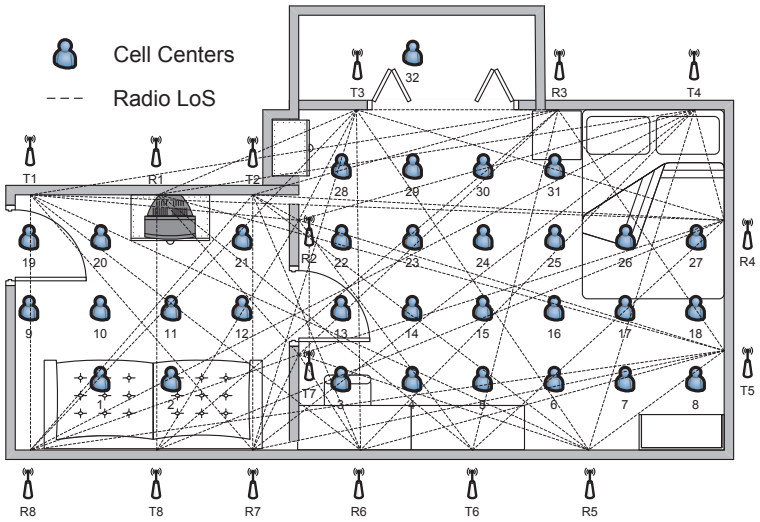
Transmitters and receivers are deployed alternatively one by one along the periphery of the wall depicted in Figure 2.7(b). Eight transmitters and eight receivers provide 64 independent radio links in total. We virtually partition the room into 32 cells, each roughly $0.75 \times 0.75\ m$ in size. We choose 0.75 m because it is the typical walking step size for adults.

Data Collection: Our method consist of the following two phases:

- Off-line training phase. In the training phase, we will construct the radio map of the room by making 100 measurements in each cell (10 seconds) to determine the RSS footprint. We consider two training strategies. In the first case (*training case A*), the subject will stand at the center of each cell and spin around so that the resulting training data will focus on the cell center but involve different orientations. In the second case (*training case B*), the subject will walk randomly within the cell. Thus, the resulting training data treat all the voxels within that specific cell uniformly and includes all possible orientations.

- On-line testing phase. In the testing phase, the subject (who is different from the subject in the training phase in height and weight) will appear in a random location with a random orientation. In our experiments, we have 100 test locations in each cell, resulting in a total of 3200 test locations. Among the 100 test locations within each cell, 25 of them are the cell center, 25 of them are 0.13 m from the center, 25 of them are 0.25 m from the center, and the other 25 are 0.38 m from the center. For each test location, we take 10 RSS measurements and compute the median value for all the 64 radio links.

(a)



(b)

Figure 2.7: In (a), we show a rather cluttered one-bedroom deployment region. In (b), we show the experimental topology. The one-bedroom deployment region is partitioned into 32 cells. The center of each cell is marked in the picture. Eight transmitters and eight receivers are deployed. We only show the 64 LoS links here.

2.4.3 Deployment Cost

Unlike [71, 85], our localization algorithm does not require prior information about the locations of all the radio nodes. Transmitters and receivers can be deployed at random locations. This property enables that PC-DfP can be applied in an environment with no changes to the existing infrastructure. In our experiments, it takes 10 seconds to collect 100 training measurements. Even considering the extra overhead of moving and turning, 30 seconds are sufficient for each cell. **Usually we spend around 15 minutes training the whole deployed region**. Given 32 cells and 100 RSS training measurements for each cell in a 64 dimensional space, it takes 0.044 seconds to estimate the parameters of the classification algorithm, and takes only 0.007 second to estimate the subject location.

Overall, the runtime cost of our method is rather modest. In the results section, we discuss ways of further reducing this cost while maintaining high localization accuracies.

2.5 Results

In this section, we first discuss performance metrics, and then present detailed experimental results.

2.5.1 Performance Metrics

The objective of a localization system is to maximize the likelihood of correctly estimating a subject's location and minimize the average distance between the estimated location and the actual location. For a specific test i, suppose a subject is actually located in cell y_i, and the estimated cell ID is $\hat{y}_i$ by *PC-DfP*. Further suppose we have N_{tst} tests. We thus define the following performance metrics:

- *Cell Estimation Accuracy* is defined as the ratio of successful cell estimations with respect to the total number of estimations, i.e., $\sum_{i=1}^{N_{tst}} I(y_i = \hat{y}_i)/N_{tst}$. In our system, we consider a test successful if the estimated cell is the same as the occupied cell.

If the subject is located on the shared boundary between two adjacent cells, the test is considered successful if the estimated cell is either one of the two bordering cells.

- *Average localization error distance* is defined as the average distance between the actual point location of the subject and the estimated point location (i.e., the center of the estimated cell).

Table 2.1 summarizes the important parameters used in our experiments. To reiterate, our experiments were conducted in a one-bedroom apartment with the total area of 5×8 m, which is divided into 32 cells (size of each cell being 0.75×0.75 m). We have 8 transmitters and 8 receivers, resulting in 64 links in total. We note that this number can be made smaller with minimal impact on our localization results. We also note that we anticipate a reasonably large number of sensors/radio devices will be existing in a "smart" home environment. In the training phase, the first author stood in each of these 32 cells, and took 100 RSS measurements. The entire training was finished within 15 minutes by one person.

2.5.2 Comparing Three Discriminant Analysis Methods

We first compare the results of the three discriminant analysis methods, namely MED, LDA, and QDA. In this set of experiments, the radio frequency is set to 433.1 MHz, and we adopt the training case A. The results are summarized in Table 4.1.

We observe that LDA performs the best among the three. We expected LDA to outperform MED because it takes into consideration the property of radio propagation. The fact that QDA is the worst of all three, however, is somewhat counter intuitive.

Parameter	Default value	Meaning
K	32	Number of cells
L	64	Number of radio links
N_{trn}	100	Number of training RSS vector per cell
N_{tst}	100	Number of testing RSS vector per cell

Table 2.1: System parameters.

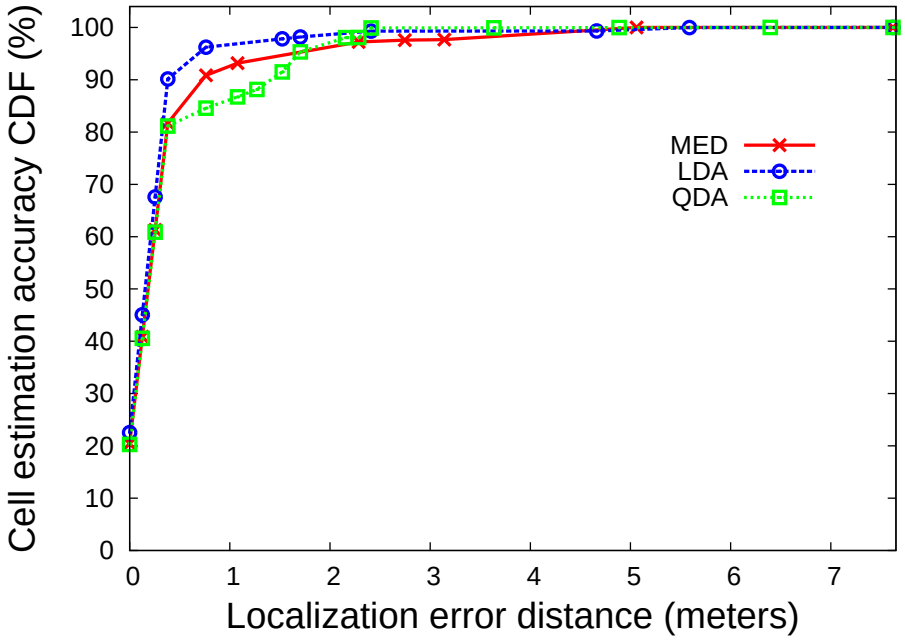


Figure 2.8: Comparing the CDF of error distances with different discriminant analysis algorithms (MED, LDA, and QDA) at 433.1 MHz.

After some deliberation, we find out the reason is that QDA requires the estimation of separate covariance matrices for each class, which can lead to over-fitting, especially with a rather limited sample size. The same trend is demonstrated in Figure 2.8 through the CDF of error distances for the three methods. (We note that QDA does have a slightly shorter tail than LDA.)

In the rest of the performance section, we will thus focus our discussion on LDA.

Discriminant Analysis Method	Cell Estimation Accuracy (%)	Average Localization Error Distance (m)
MED	81.7	0.55
LDA	90.1	0.44
QDA	81.1	0.53

Table 2.2: Comparison of the three discriminant analysis methods: MED, LDA, and QDA in training case A.

2.5.3 Mitigating Multipath Effect

We have mentioned that the multipath effect has an adverse impact on indoor localization, and in this chapter, we have devised approaches to mitigate its impact for improved localization results. Specifically, due to multipath, when a subject moves around, we will observe large and abrupt RF variations, even within a cell. Therefore, accurately estimating cell ID based upon the observed RF readings becomes a daunting task. To mitigate this impact, we take the following measures to smooth out the RF variations within a cell.

First, we operate our radios at the unlicensed frequency of 433.1 MHz instead of 909.1 MHz. Intuitively, the wavelength at 433.1 MHz is larger than that at 909.1 MHz, and thus the RF signal has a smoother variation when the subject is moving. We have conducted an experiment to demonstrate this idea. Figure 2.1(a) shows the experimental setup, and Figures 2.1(b) and Figures 2.1(c) shows the RF variation is much smaller at 433.1 MHz than at 909.1 MHz.

Second, in the training phase, instead of standing still at a specific point within a cell and using the measurement at that point to represent the entire cell (as in training case A), we make random movements within that cell, take multiple measurements, and use them collectively for classification, as in training case B in Section 2.4. In this way, we sample the data for all the voxels with different orientations to average out the multipath effect within each cell.

Table 2.3 summarizes the LDA results with and without these two optimizations. We also varied the test location in these experiments. In general training case B gives better cell estimation accuracies than training case A. Within each training case, radio frequency of 433.1 MHz gives better results than 909.1 MHz with the node layout shown in Figure 2.7(b). *In summary, our cell estimation accuracy is as high as 97.2% with*

	433.1 MHz	909.1 MHz
Training case A	90.1%	82.9%
Training case B	97.2%	93.8%

Table 2.3: LDA cell estimation accuracies improve when the radios work on 433.1 MHz, and adopt the training case B.

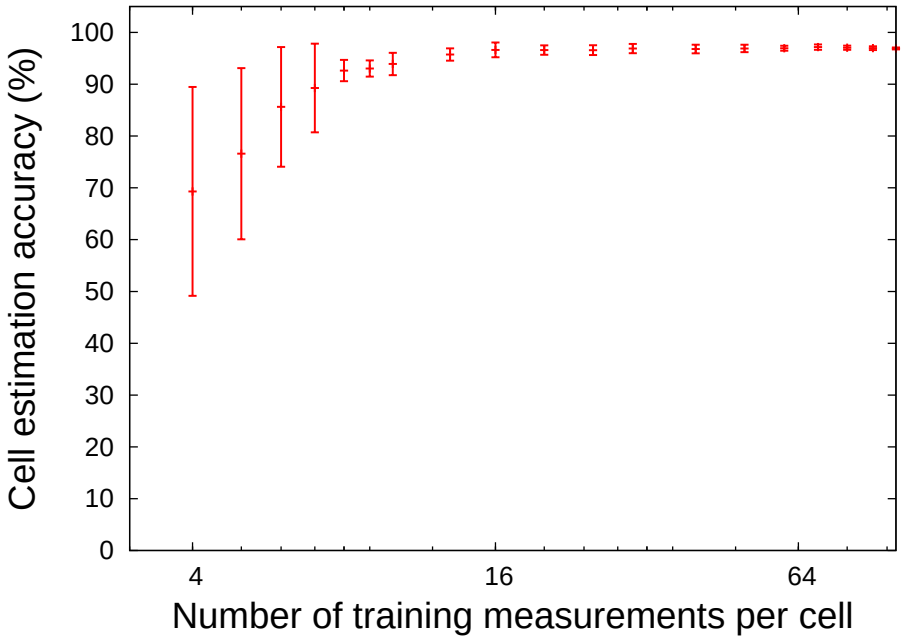


Figure 2.9: Cell estimation accuracy with 95% confidence interval error bar versus the number of training measurements.

the average localization error distance of 0.36 meters.

2.5.4 Reducing Training/Testing Overhead

Here we investigate methods for reducing the computing overhead for our algorithm. In this study, we formulate the localization problem as a classification problem that involves a training phase and a testing phase. Suppose we have N training data of L dimensions and K classes, where N is the number of measurements taken in each cell in the training phase, L is the number of radio links in the environment, and K is the number of the cells in the environment. In our default setting, we have $N = 100$, $L = 64$, and $K = 32$.

For LDA, the algorithmic complexity is $O(KNL + K^3)$ in the training phase and $O(KL^2)$ in the testing phase. As K is fixed in our algorithm, we can try to use a smaller N and/or L to reduce the overhead.

First, we look at the possibility of having a smaller N, i.e., fewer training samples. Figure 2.9 shows the localization results with different training data sizes. We observe that we achieve a cell estimation accuracy of 90% by using 16 training measurements

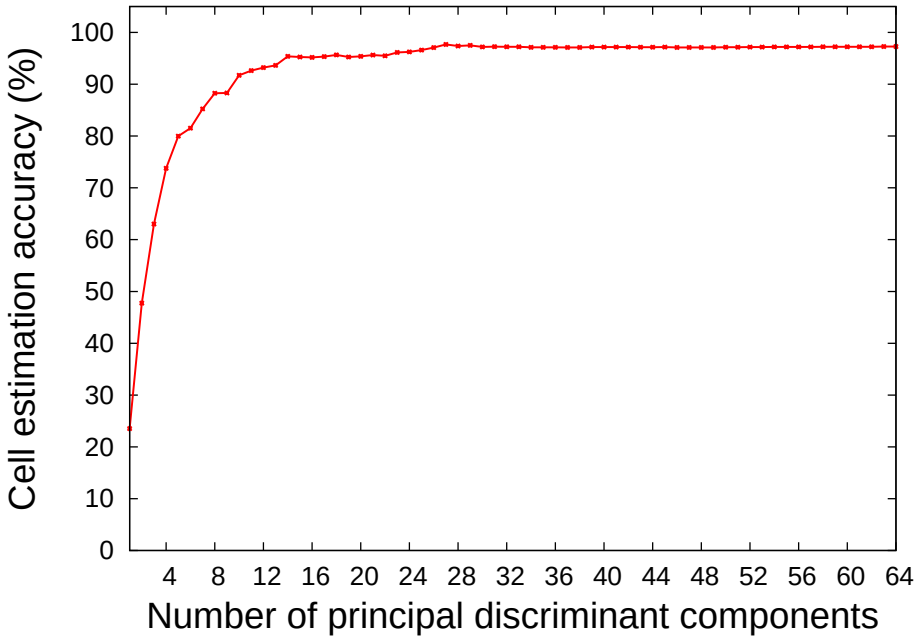


Figure 2.10: Cell estimation accuracy as a function of the number of most important principal discriminant components.

in each cell, and achieve a cell estimation accuracy as high as 90% by only using 8 training measurements in each cell. This will lead to a significant reduction of the training overhead.

Next, we look at the possibility of having a smaller L, i.e., smaller data dimensions. To do so, we adopt the optimization technique discussed in Section 2.3 to select the principal discriminant components for classification purpose. Figure 2.10 shows that we can achieve the same level of cell estimation accuracy when using only the first 28 principal discriminant components. Such a reduction on data dimension can lead to significant improvement on computation efficiency. If we are willing to relax the requirements for the cell estimation accuracy from 97% down to 90%, then choosing the 10 most principal discriminant components will be sufficient.

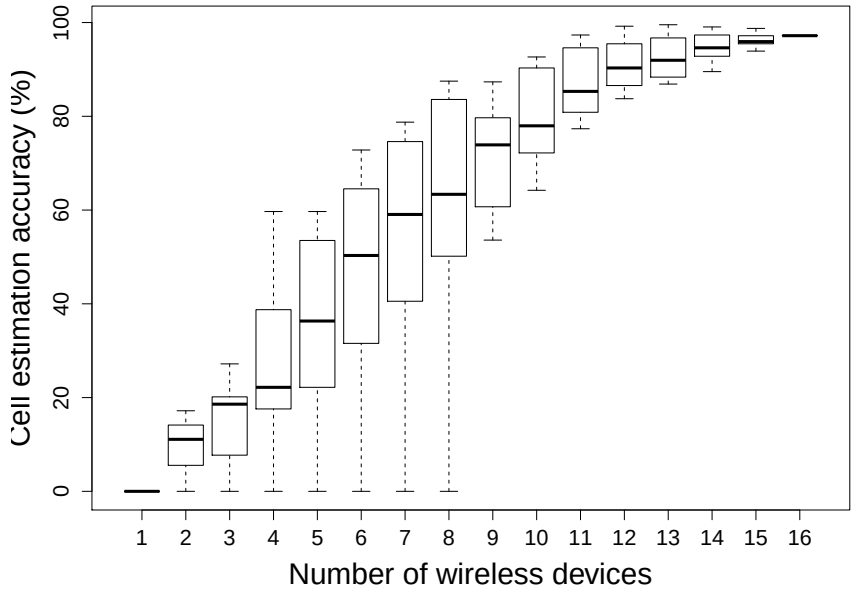


Figure 2.11: Boxplot of cell estimation accuracy versus the number of wireless devices that are used. For a given number, we show all the possible combinations.

2.5.5 Localizing Subjects with Minimum Number of Radio Nodes

Next, we need to test whether our system can still function if we lose one or more radios. In the experiments, we use a subset of the radio nodes and derive the corresponding localization results, and investigate at what point the cell estimation accuracy will drop below a tolerable level. For example, if we would like to find out the results using 10 radio devices out of the default 16 (8 transmitters and 8 receivers), we would randomly remove 6 devices, and plot the localization results for all the possible combinations of transmitters and receivers.

These results are shown in Figure 2.11. We find that our algorithm can deliver a cell estimation accuracy of 90% when we remove 3 transmitters and 3 receivers in the process. Finally, we note that our system can achieve an even better accuracy (than having all 16 nodes), 99.4%, when three particular devices (i.e., T7, R4 and R6) are removed. Note that we do not reposition the remaining nodes to optimize the results, so this is an overestimate of the number of nodes needed for a given accuracy. Optimizing localization results by systematically removing radio devices (as well as

the corresponding links) is a topic for further investigation.

2.5.6 Using the Same Training Set Over a Long Testing Period

All the results shown above have the testing phase done within three days after training the system. In reality, we are also interested in knowing how well our system will perform if the testing occurs much later in time, which could lead to performance degradation because of changes in the environment or drift in the radio. Different subjects or changes in the same subject could also affect the results.

All the above factors can change the relevance of the original RSS calibration and training. Thus, we need to find an effective correction technique to extend the accuracy of an original calibration over weeks or months. The basic idea is that before each experiment, be it training or testing, we always collect the environmental RSS vector RSS_E when the room is unoccupied. We refer to this vector as RSS_E^{trn} and RSS_E^{tst} for the training and testing phase respectively. This information provides the correction basis for the test data. We can determine when to collect RSS_E^{tst} based upon the subject's life style. For example, it can be collected at noon if he/she works regularly, or at midnight if he/she stays home most of the time.

Using the environmental RSS vector, we propose the following two correction approaches:

- Naive correction: For a simple correction of change over time, we first compute the pairwise difference between the RSS_E^{trn} and RSS_E^{tst}, and record the vector as RSS_E^{bias}. Then we add this bias vector to each RSS vector as the compensation and construct the new test data.

- Truncated correction: We compute RSS_E^{bias} as with naive correction and set an empirical threshold τ. Then we compare the ith entry ${RSS_E^{bias}}_i$ with τ for $i \in 1, ..., L$. If $\mid {RSS_E^{bias}}_i \mid \geq \tau$, then we eliminate that feature (link) from both training data and test data. Otherwise, we compensate the test data for that feature as in naive correction. The rationale behind this approach is that we want to eliminate

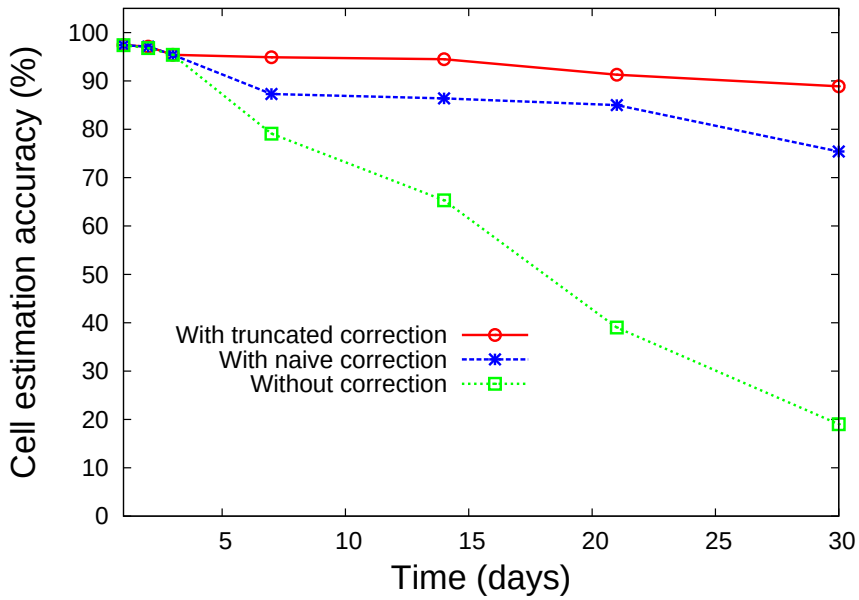


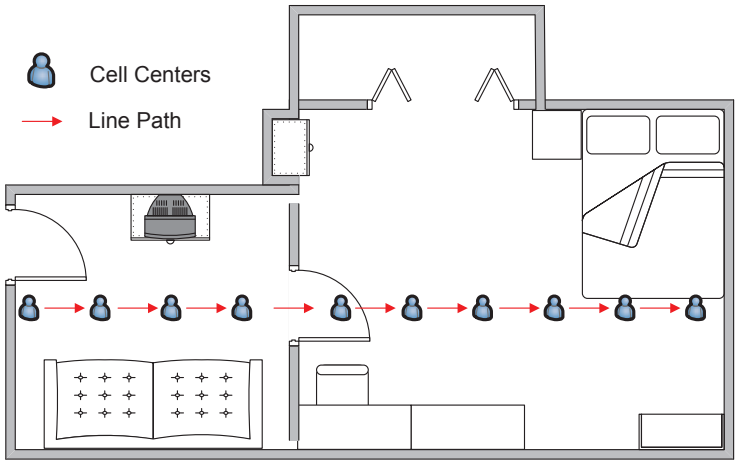
Figure 2.12: Cell estimation accuracies over one month after the training with different correction approaches.

those links that have experienced a large variation due to environmental instabilities. Since our earlier results (Figure 2.11) show that our system is robust against missing a few links, we believe removing these links with large fluctuation will not significantly degrade the performance.

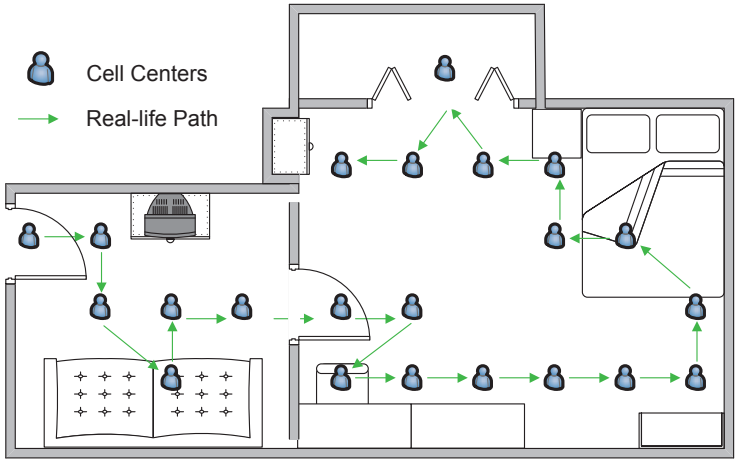
We summarize the results in Figure 2.12. In the case without correction, we do not subtract the environmental RSS from the training/test data. The results show that cell estimation accuracies drop significantly one week after training the system without any correction. With naive correction, we can achieve a cell estimation accuracy of 80% after one month, and truncated correction provides 90% cell estimation accuracy after one month, which is the best among all three.

2.5.7 Tracking a Moving Subject

Our approach can also be used to track a moving subject. In this set of experiments, the subject moves in the apartment, and we try to estimate which cells he passes during the movement. We choose the longest straight line path and a zigzag path as representatives to test PC-DfP's tracking performance. Specifically, the subject adopted



(a)



(b)

Figure 2.13: Two mobility paths: (a) a line path, and (b) a real-life path.

the following two mobility patterns: (1) *line path*, in which the subject walked along a straight line at an average speed of about 3 meters per second (illustrated in Figure 2.13(a)), and (2) *real-life path*, in which the subject followed a path similar to the path taken in his real life, e.g., he might choose to walk to the bed and lie on the bed for a few seconds, and then walk to the couch and sit down on the couch for a few seconds (illustrated in Figure 2.13(b)). When the subject moved in the room, we continued to take measurements and estimated which cell he occupied.

We show the localization results in these two cases in Table 2.4. As expected, when the subject moves along a line path, he can be localized almost as well as when he is stationary, with an cell estimation accuracy of 99.1% and a localization accuracy of 0.3 m. The results for the real-life path are slightly worse (cell estimation accuracy being 91.1%) because there are more complicated movements including walking, lying down, getting up, and sitting. As a result, more uncertainties are introduced. In particular, the cell estimation accuracy is 86.1% when subject is moving and 98.6% when the subject stays still on bed, chair or sofa. We, however, would like to point out that the average localization error distance in this case, 0.37 m, is still rather good. We note that different paths will lead to varying accuracies as different cells have exhibited different classification accuracies.

In this study, we directly apply our approach to the mobile case without any modification. In our next step, we would like to investigate more sophisticated methods such as taking into consideration the trajectory information.

2.5.8 Localizing Multiple Subjects When Subject Count Is Known

Next, we extend our method to localizing multiple subjects that coexist in a room if we know the number of subjects. Here, we do not need to do any additional training,

Different Mobility Path	Cell Estimation Accuracy (%)	Average Localization Error Distance (m)
Line	99.1	0.3
Real-life	91.1	0.37

Table 2.4: Localization results with two different mobility paths.

and the original training data is sufficient.

In our method, we plug the measured data into the classifier, and retrieve the class label which gives the maximum value from the discriminant functions to estimate the cell number. Similarly, to localize n subjects, we just simply pick n class labels which have the first n maximum values. For multiple subjects localization, we define the cell estimation accuracy as the ratio of the number of the occupied cells that are correctly estimated to the number of subjects. For instance, if there are three subjects and only two of their three cells are correctly estimated, then the cell estimation accuracy will be 66.7%.

We perform 32 independent tests, and show our results in table 2.5. As expected, the cell estimation accuracy decreases when the number of people increases because more people will cause a higher degree of uncertain interference with radio signals.

2.5.9 Deploying Our Method to a Larger Office Environment

We have shown that our localization method works well in a home environment where radio devices are installed on the walls. Next, we apply our method to a larger office environment to show that it can easily scale to a different setting. In our experiments, we deploy 13 transmitters and 9 receivers in the first author's office, which is 10×15 m in size. In such an environment, localizing subjects at a 0.75×0.75 m cell granularity is not needed; instead, a cubicle-size cell should be sufficient. Thus, we can still partition the deployed area into 32 cells, as shown in figures 2.14(a) and 2.14(b). This deployment has two main differences compared to our original deployment: heterogeneous cell sizes and random radio positions (i.e., not always on the walls). Using the same method, our cell estimation accuracy is 93.8% and the average localization error

Number of People	Cell Estimation Accuracy (%)	Average Localization Error Distance (m)
1	97.2	0.36
2	89.5	0.82
3	83.5	0.89

Table 2.5: Localization results with respect to number of people in the room when the number is known.

distance is 1.4 m. This degradation compared to the performance in the one-bedroom apartment can be explained as follows. Intuitively, a larger cell involves more voxels, which result in a large variance for each class. Therefore, for all the classes, there is a higher probability that each pair-wise class will have a larger intersection area, which leads to more classification error.

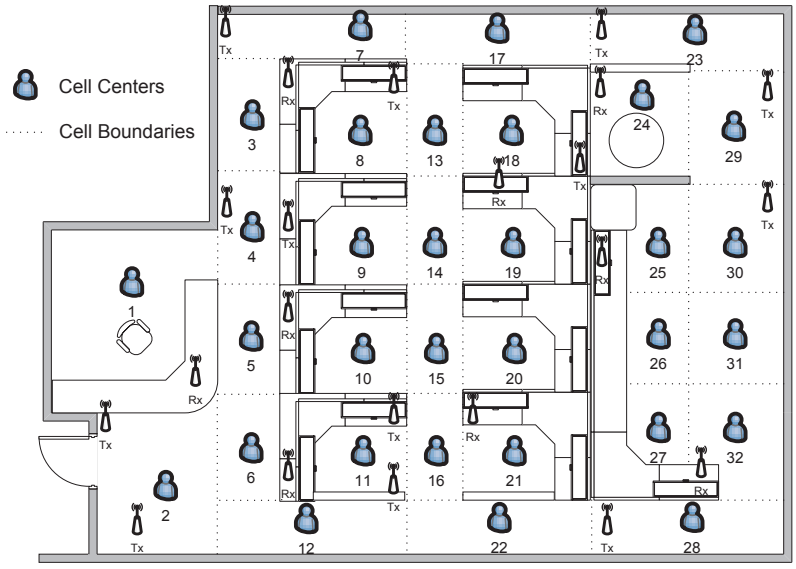
2.6 Related work

In this section, we discuss the related work in device-free passive localization (for stationary subjects) and tracking (for mobile subjects).

Device-free Passive (DfP) Localization: Several DfP approaches have been proposed in the literature. In [83, 62], DfP localization is done through fingerprint matching. A passive radio map is constructed during the training phase by recording RSS measurements with a subject standing at pre-determined locations. During the testing phase, the subject appears in one of these locations, and the system can match the observed RSS readings to the RSS readings from one of the trained locations based upon minimum Euclidean distance. Our method shares the same philosophy with [83, 62] in that multipath is so complex that we cannot understand the direct relationship between a subject's location and the radio signal changes. Instead, we have to train the system first. However, minimum Euclidean distance is shown not to be as efficient as LDA in classification in our study. Further, we have taken special care in the training phase to minimize the RF signal variation within short distances to mitigate the multipath effect. These measures are based upon our in-depth understanding of the radio propagation properties and can lead to much improved localization results.

Radio tomography imaging [71] is a technique to reconstruct the tomographic image for localizing device-free subjects. Here, the authors assume that the relationship between a subject' location and the radio signal variation can be mathematically modeled. In [71, 11], based upon the shadowing effect (RSS is attenuated when the LoS is blocked) caused by the subject, a linear attenuation model and a Sequential Monte Carlo model are proposed respectively. This technique is unlikely to fare well in a

(a)



(b)

Figure 2.14: In (a), we show the first author's lab in which we deployed our system. In (b), we show the experimental topology. The office deployment region is partitioned into 32 cubicle-sized cells. Thirteen transmitters and nine receivers are deployed. We show the cell boundaries in this plot.

cluttered indoor environment because we observed that a person blocking the LoS can only attenuate the RSS with a 50% probability (Section 2.2).

Device-free Passive Tracking: Several techniques have been proposed to track a moving subject in a passive fashion. In [85, 86], a grid sensor array is deployed on the ceiling for the tracking purpose. An "influential" link is one whose RSS change exceeds a empirical threshold. The authors calculate a subject's location based upon the observation that these influential links tend to cluster around the subject. This work is extended in [84] with triangle sensor array deployment and training information. In VRTI [72], the authors leverage the RSS dynamics caused by the moving subject to generate a radio tomographic imaging for tracking.

Finally, we would like to point out not only fingerprint-based schemes (including ours) need a training phase, but other schemes such as radio tomography and grid sensor array also need a training phase to determine a suitable threshold value to detect if a subject is on the radio LoS.

2.7 Conclusion

In this chapter, we present the design, implementation, and evaluation of a device-free passive localization method based on probabilistic classification. We compare three discriminant analysis techniques and find that linear discriminant analysis (LDA) yields much better localization results than minimum Euclidean distance (MED) and quadratic discriminant analysis (QDA). We also propose ways of mitigating the error caused by multipath effect for better localization results, and approaches for correcting training data to facilitate tests much later than the original training. We evaluate our method in a real home environment, rich in multipath. We show that our system can successfully localize a subject with 97% cell estimation accuracy within 0.36 m error distance. Through detailed experiments, we demonstrate that our method can achieve a basic accuracy of over 97%. More importantly, it can maintain an accuracy of over 90% with a substantial reduction in number of radio devices (from 16 down to 10), with far fewer training samples (from 100 to only 16 per cell), or the use of a training set taken a

month before testing. In addition, the basic system, without modification, can also be used to track a moving subject or localize multiple subjects. Though originally tested in a small apartment, it performs well in a larger commercial office space.

Chapter 3

SCPL: Device-free Multi-Subject Counting and Localizing Using Radio Signal Strength

In this chapter, we present SCPL, a framework aiming to count and localize multiple people using radio signal strength at one time.

3.1 Introduction

Ambient Intelligence (AmI) envisions that future smart environments will be sensitive and responsive to the presence of people, thereby enhancing everyday life. Potential applications include eldercare, rescue operations, security enforcement, building occupancy statistics, etc. The key to enable these ubiquitous applications is the ability to localize various subjects and objects in the environment of interest. Device-free passive (DfP) localization has been proposed as a way of detecting and tracking subjects without the need to carry any tags or devices. It has the additional advantage of being unobtrusive while offering good privacy protection. Over the past decades, researchers have studied ways of tracking device-free human subjects using different techniques such as camera [31], capacitance [67], pressure [49], infrared [16] and ultrasonic [23]. However, they all suffer from serious limitations such as occlusion [31, 16], high deployment cost [49, 67] or short range [23].

Radio frequency (RF)-based techniques have the advantages of long-range, low-cost, and the ability to work through non-conducting walls and obstacles Several RF-based DfP localization techniques have been proposed in [83, 85, 50, 43, 72, 11, 73, 78, 25], and these approaches observe how people disturb the pattern of radio waves in an indoor space and derive their positions accordingly. To do so, they collect training

data to profile the deployed area, and form mathematical models to relate observed signal strength values to locations. DfP algorithms can be broadly categorized into two groups: *location-based*, and *link-based*. Location-based DfP schemes collect a radio map with the subject present in various predetermined locations, and then map the test location to one of these trained locations based upon observed radio signals, which is also known as fingerprinting, as studied in [83, 78]. Link-based DfP schemes, however, capture the statistical relationship between the received signal strength (RSS) of a radio link and whether the subject is on the Line-of-Sight (LoS) of the radio link, and consequently determine the subject's location using geometric approaches [85, 50, 11, 25].

Recognizing that merely tracking an individual might not be sufficient for typical indoor scenarios, researchers have been pushing a great amount of effort towards scaling to multiple device-free subjects, such as in [86, 84, 45, 73, 78, 46]. They observe the change of RSS mean or variance and propose different tracking algorithms. The common thing missing is that the number of subjects is known, which is a strong assumption. In addition, in cluttered indoor environments, subjects can cause collective nonlinear fading effects, which might significantly degrade the tracking performance and is not explicitly treated in the work above. On the other hand, location-based schemes can be straightforward but prohibitive due to the exponential increase in the training overhead if we need to profile the system with different combinations of these subjects.

In this study, we propose and evaluate an efficient DfP scheme for tracking multiple subjects using the training data collected by a single subject to avoid expensive training overhead.

Our algorithm consists of two phases. In the first phase, we *count* how many subjects are present using successive cancellation in an iterative fashion. In each iteration, we detect whether the room is empty. If it is not empty, we identify the location for one subject, and then subtract her impact on the RSS values from the collective impact measured in the experiment. Care must be taken when subtracting a subject's impact as the change in the RSS values caused by multiple subjects at the same time is smaller than the sum of RSS changes from each individual subject. In order to compensate

for this, we need to multiply a coefficient to a subject's impact and then perform subtraction. The coefficient is specific to the subject's location as well as the link under consideration.

In the second phase, we localize the subjects after their number is known. We partition the deployment area into cells and represent a subject's location using its cell number. We formulate the localization problem as a conditional random field (CRF) by modeling indoor human trajectories as a state transition process and considering mobility constraints such as walls. We then identify the cells occupied by these subjects simultaneously. Since our counting process is sequential and our localization process is parallel, we call our algorithm *SCPL*.

We have tested SCPL in two indoor settings. The first setting is an office environment consisting of cubicles and narrow aisles, which is partitioned into 37 cells. We used the 13 transmitters and 9 receivers that were deployed for some earlier projects. The second setting is an open floor indoor environment, which is partitioned into 56 cells and deployed with 12 transmitters and 8 receivers. In the training phase, we measured the RSS values using a single subject. In the testing phase, we had four subjects with different heights, weights and gender, and designed four different real life office scenarios. These scenarios all had periods of time when multiple subjects walked side by side and thus had overlapping trajectories. We can count the number of subjects accurately, with a 88% counting percentage when the subjects were not walking side by side, and a 80% counting percentage when they were.

Our localization results have good accuracies, with a average error distance of 1.3 m considering all the scenarios. We find that it is beneficial to consider indoor human movement constraints according to the floor map when localizing moving subjects and demonstrate 24% improvement on average compared with no floor map information provided.

Our technique, SCPL, is unique in at least four contributions: (i) to our knowledge, it is the first work to systematically perform simultaneous counting and localization for up to four device-free subjects (moving or stationary) in large-scale deployments only using RF-based techniques; (ii) we designed a set of algorithms to count and

localize multiple subjects relying on the calibration data collected by only a single individual; (iii) We also use plausible trajectory constraints (e.g. not walking through walls) based on floor map information, and integrate this information into the radio calibration data to further improve the tracking accuracies; and (iv) we recognize the nonlinear fading effects caused by multiple subjects in cluttered indoor environments, and design the algorithms to mitigate the resulting error.

The rest of the chapter is organized as follows. In Section 3.2, we discuss the applications that benefit from passive localization as well as our solution framework. Our solution consists of two phases, counting the number of subjects (in Section 3.3) and localizing the subjects (in Section 3.4). Then we describe our experimental setup in Section 3.5 and our detailed results in Section 3.6. We discuss the limitation and future direction of our work in Section 3.7 and review the related work in Section 4.7. Finally, we provide the concluding remarks in Section 3.9.

3.2 Background

Before presenting our SCPL algorithm, we first discuss potential applications and the formulation of the problem.

3.2.1 Applications that Can Benefit from Passive Localization

Passive localization can find application in many important domains. Below we give a few examples:

Elderly/Health Care: Elder people may fall down in their houses for various reasons, such as tripping, momentary dizziness or overexertion. Without prompt emergency care, this could lead to life-threatening scenarios. Using trajectory based localization information, DfP can perform fall detection quickly because the monitored subject will remain in an unusual location for a long period of time.

Indoor Traffic Flow Statistics: Understanding patterns of human indoor movement

can be valuable in identifying hot spots and corridors that help energy management and commercial site selection. DfP provides a non-intrusive and private solution to capturing indoor locations.

Home Security: DfP based home security is a major improvement over camera-based intrusion detection because it can not only detect the intrusion, but also track the intruders.

3.2.2 Problem Formulation

To solve the passive multi-subject localization problem, we adopt a cell-based fingerprinting approach, similar to the one discussed in [78].

Before we address the multi-subject problem, let us first look at how we localize a single subject. We first partition the deployed area into K cells. In the training phase, we first measure the ambient RSS values for L links when the room is empty. Then a single subject appears in each cell, walks randomly within that cell and takes N RSS measurements from all L radio links. By subtracting the ambient RSS vector from the collected data, we have a profiling dataset $\mathcal{D}$. $\mathcal{D}$, a $K \times N \times L$ matrix, quantifies how much a single subject impacts the radio RSS values from each cell. Having this profiling dataset $\mathcal{D}$, *we model the subject's presence in cell i as state S_i* and thus $\mathcal{D} = \{\mathcal{D}_{S_1}, \mathcal{D}_{S_2}, ..., \mathcal{D}_{S_K}\}$. In the testing phase, we first measure the ambient RSS values when the room is empty. Then a subject appears in a random location, and measures the RSS values for all L links while making random moves in that particular cell. Then we subtract the ambient RSS vector from this measured data, and form an RSS vector, O, which shows how much this subject impacts the radio links from this unknown cell. Based on $\mathcal{D}$ and O, we can run classification algorithms to classify the cell number of the unknown cell, thus localizing the subject.

Next we discuss how we extend the same framework to formulate the multi-subject localization problem. In the training phase, our objective is to still *use a single subject's training data* to keep the training overhead low. Taking the training data for different number of subjects will lead to prohibitive overheads, which we will avoid. In the

testing phase, multiple subjects appear in random cells, sometimes in the same cell, and we measure the RSS values for all the radio links. We calculate O in the same way as in the single-subject case.

To calculate the locations for these subjects, we need to go through two phases. In the first phase, we identify the number of subjects that are present simultaneously, C, which we call the *counting* phase. In the second phase, we identify in which cells are these C subjects, which we call the *localizing* phase. Please note that subjects are not stationary, but they move around within the deployed area.

3.3 Counting the Number of Subjects

In this section, we first provide empirical data to help the readers understand the impact of having multiple subjects on the radio signals, especially nonlinear fading effect, and then describe our sequential counting algorithm.

3.3.1 Understanding the Impact of Multiple Subjects on RSS Values

Let us first understand the relationship between a single subject's impact on the room RSS level and multiple subjects' impact. In particular, we would like to find out whether the relationship is linear.

As shown in previous studies such as [83, 85, 71, 11, 78], the RSS level of a radio link changes when a subject is near its Line-of-Sight (LoS). Based on this observation, we make a simple hypothesis: *more subjects will not only affect a larger number of spatially distributed radio links, but they will also lead to a higher level of RSS change on these links.* If this is true, we can infer the number of subjects that are present from the magnitude of the RSS change that we observe in the deployed area. We use the sum of the individual link RSS change to capture the *total energy change* in the environment as

$$\gamma = \sum_{l=1}^{L} O^l,$$

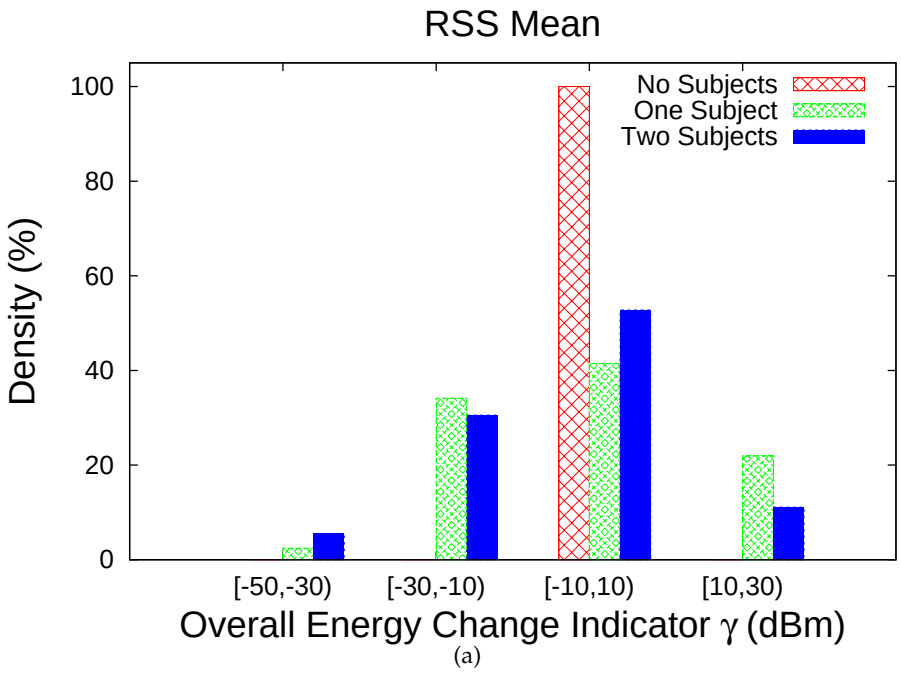
where O^l is the RSS change on link l.

Next we look at how to capture the RSS change of link l. A straightforward metric is to subtract the mean ambient RSS value for link l (when the room is empty) from the measured mean RSS value for link l, the result of which is referred to as *RSS mean difference*. RSS mean difference is a popular metric that has been used in several studies, e.g., as seen in [83, 71, 11, 78]. However, upon deliberation, we find that RSS mean difference is not suitable for our purpose, mainly because the value is not always positive. Due to the multi-path effect, the presence of a subject does not always weaken a link, but sometimes, it may actually strengthen a link! As a result, the RSS mean difference can be negative. In this case, summing up each link's RSS mean difference does not lead to the correct total energy change in the environment because their values may cancel out each other. To address this issue, we thus propose to use *absolute RSS mean difference* which has a more compact data space than RSS mean when a cell is occupied.

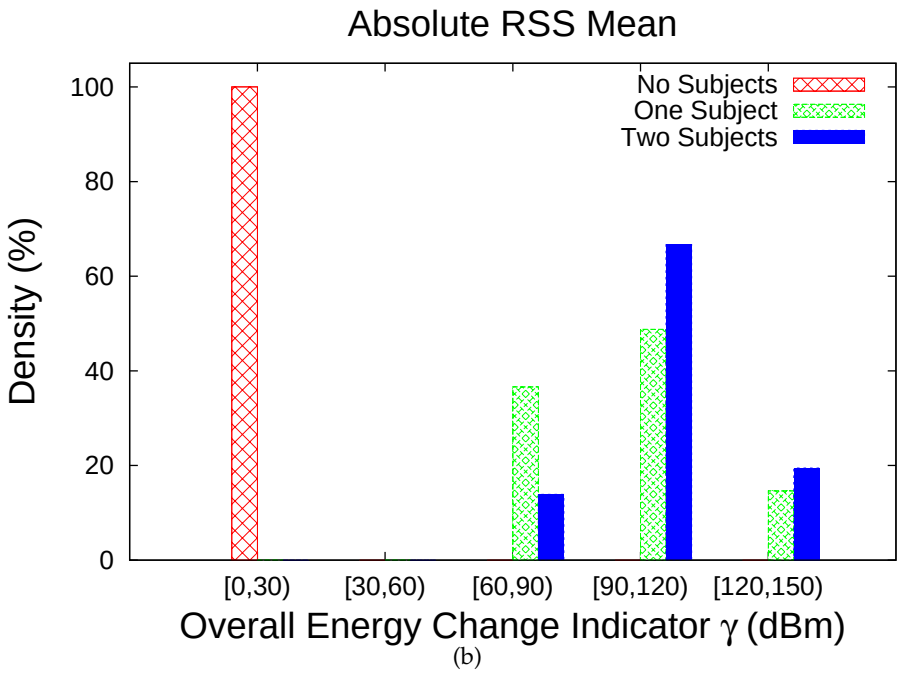
Our experimental results confirm that the absolute RSS mean difference is a more suitable metric. In this set of experiments, we collect the RSS values when there are 0, 1 and 2 subjects who make random movements (with pauses) in the deployed area. We compute the corresponding γ value by using both RSS mean difference and absolute RSS mean difference, and plot their histograms in Figures 3.1(a)-(b) respectively. In Figure 3.1(a), when the room is empty, we observe γ values $\in [-10, 10)$ which means the overall energy level is rather stable. However, with 40% to 50% of chances, we still observe $\gamma \in [-10, 10)$ when subjects are present. This is because individual RSS mean differences can cancel out each other, and thus their sum is not a good indicator of the total energy change caused by having multiple subjects.

Absolute RSS mean difference is a better metric, as shown in Figure 3.1(b). The γ value when there are two subjects is statistically greater than the γ value when there is only one subject. As a result, in the rest of this chapter, unless explicitly noted, we use absolute RSS mean difference as the metric to capture the RSS change in the environment. Finally, we note that the γ value alone is inadequate to distinguish between one or two subjects.

By looking at the two-subject data more carefully, we can further separate them



(a)



(b)

Figure 3.1: In terms of overall energy change indicator γ, (a) "RSS Mean", for zero, one, and two subjects. (b) "Absolute RSS Mean" for the same measurement shows better discrimination between zero and more than zero subjects.

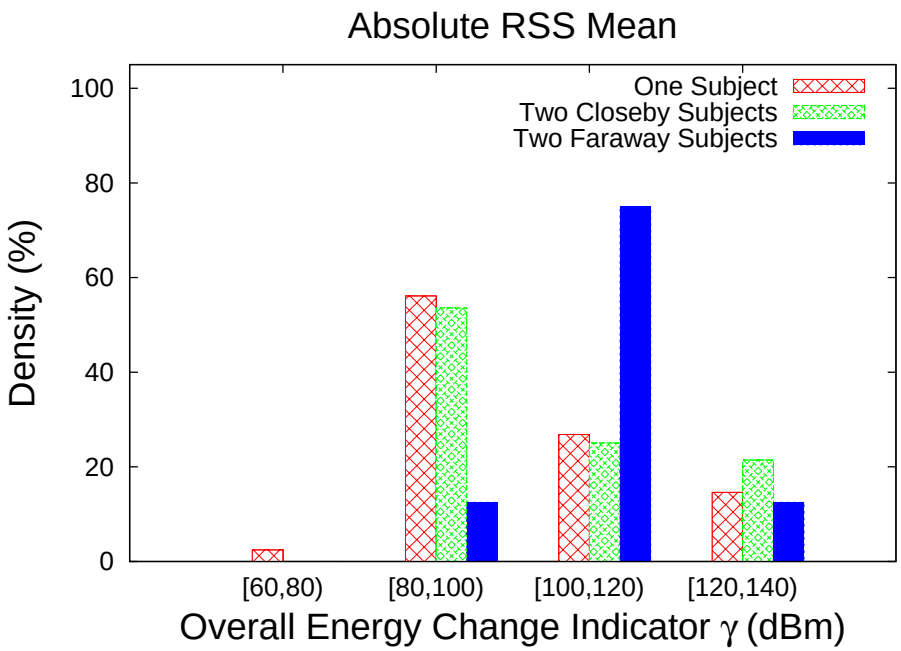


Figure 3.2: In terms of overall energy change indicator γ, Two subjects separated by more than 4 meters are clearly distinguishable from one subject.

into two groups based on the distance between the subjects. If the distance is more than 4 meters (we choose this threshold from the data sets), we call the two subjects *faraway*, and call the subjects *nearby* if the distance is less. We then plot the histograms of these groups in Figure 3.2. When subjects are close to each other, more links will be affected by both subjects, and fewer links are affected by only one of the subjects. Consequently, the γ value in this case will be smaller than the γ value when the two subjects are farther apart. Furthermore, we point out that the γ value when we have C subjects at the same time is smaller than the sum of the individual γ value from each subject. As a result, it is hard to distinguish having two subjects close to each other from having only one subject.

In summary, we have two main observations from these experiments. First, the absolute RSS mean difference is a suitable metric to capture the impact caused by the appearance of a subject. Second, the total energy change, γ, reflects the level of impact subjects have in the room, but we cannot rely on the value of γ alone to infer how many subjects are present because γ is not linearly proportional to the number of subjects.

3.3.2 Counting Subjects Using Successive Cancellation

We use successive cancellation to count the number of subjects. When multiple subjects coexist, it often so happens that one subject has a stronger influence on the radio signal than the rest. Thus, our counting algorithm goes through several rounds. In each round, we estimate the strongest subject's cell number in this round assuming there is only a single subject, i, and then subtract her share of RSS change from the remaining RSS vector O to obtain the new remaining RSS vector that will be used in the next round.

If this problem were linear, we could simply subtract the mean vector μ_i associated with cell i in the profiling data $\mathcal{D}$ from the observed RSS vector O. However, as shown in the previous subsection, the total impact from multiple subjects is not linear to the number of subjects – the impact observed when C subjects appear at the same time is smaller than the sum of each subject's impact if they appear one at at time. To be more precise, O is an underestimation of the linear combination of the mean values of the associated cells that we collected in $\mathcal{D}$. To address this issue, instead of subtracting μ_i directly from O, we multiply a coefficient that is less than 1 to μ_i and subtract this normalized term from O. This coefficient, however, is not uniform across all the cell and link combinations; instead, it is specific to each cell and link pair because different cells have different impacts on a link. We will then calculate the location-link coefficient matrix, $\mathcal{B} = (\beta_{i,l})$ where $\beta_{i,l}$ is the coefficient for cell i and link l.

Our algorithm to calculate the coefficient matrix $\mathcal{B}$ is detailed in Algorithm 1. The basic idea is that, for each link l, we compute the correlation between a cell pair, (i, j) with respect to link l. The two cells that both are close to a link are highly correlated with respect to this link. We use h_{ij}^l to denote this correlation[1]. Note that all the RSS values in profiling data are non-negative, and thus we have $h_{ij}^l \geq 0$. For each cell i, we

[1]Notice that we use correlation h_{ij}^l instead of correlation coefficient ρ_{ij}^l because ρ_{ii}^l will always be 1 and thus guarantee its dominance among the all the cells on all the links when the cell i is detected first, which is not true.

pivot that cell and compute the β_{il} as

$$\beta_{il} = \frac{h_{ii}^l}{\sqrt{\sum_{j=1}^{K} {h_{ij}^l}^2}}.$$

Basically, when two subjects occupy cells i and j respectively, and only one of them affects link l, they have low correlation and the value of h_{ij}^l is close to 0. On the other hand, when they both affect link l, the value of h_{ij}^l will reflect their positive correlation.

Algorithm 1: Location-Link Correlation Algorithm

input : $\mathcal{D}$- The training data collected from L links among K states/cells
output: $\mathcal{B}$ - The location-link coefficient matrix

for $l = 1 \rightarrow L$ **do**
  $h \leftarrow$ zero matrix of $K \times K$
  for $i = 1 \rightarrow K$ **do**
    for $j = 1 \rightarrow K$ **do**
      $I \leftarrow$ training data indices associated with state S_i
      $J \leftarrow$ training data indices associated with state S_j
      // Compute the link correlation
      $h_{ij} \leftarrow E\left[\mathcal{D}_{Il}\mathcal{D}_{Jl}\right]$
  for $i = 1 \rightarrow K$ **do**
    $normfactor \leftarrow \sqrt{\sum_{j=1}^{K} {h_{ij}}^2}$
    // Compute the location-link coefficient for cell i and link l
    $\beta_{il} \leftarrow \frac{h_{ii}}{normfactor}$

Once we determine the location-link coefficient matrix $\mathcal{B}$, we describe our successive cancellation based counting algorithm (shown in Algorithm 2), which can identify the subject count C from the observation RSS vector O using the profiling RSS matrix $\mathcal{D}$ collected by a single subject. We first compute γ^0's and γ^1's from the ambient RSS vector and the profiling RSS matrix $\mathcal{D}$ respectively. Then, we construct a 95% confidence interval for the distribution of γ^0's and γ^1's and refer to the associated lower and upper bounds as c_L^0, c_U^0, c_L^1, c_U^1. From the observation RSS vector, O, we first compute its γ value and then perform a presence detection: if $\gamma < c_U^0$, we claim the room is empty. Otherwise, we will claim there is at least one subject present and start to iteratively count the number of subjects using successive cancellation to finally determine

the value of C.

In each successive cancellation iteration, we do the following:

- *Presence Detection.* We first perform a presence detection by checking if $\gamma \geq c_U^1$ to find out whether there is any more subject in the room. Please note that this condition is stronger than $\gamma \geq c_U^0$, and we will take care of the last iteration separately. If the presence detection returns a 'yes', we increment the detected subject count C, and go to the next step. Otherwise, we end the algorithm.

- *Cell Identification.* If there is a subject in this iteration, we estimate the occupied cell q by

$$q = \underset{i \in \mathcal{S}}{\operatorname{argmax}} P(O|S_i),$$

  where $\mathcal{S}$ is the set of remaining unoccupied cells.

- *Contribution Subtraction.* Next, we cancel the impact of this subject from cell q by subtracting $\mu_{ql} \cdot \beta_{ql}$ from O^l for each link l.

In the last round, we simply check if $\gamma < c_U^1$, which actually relax the lower bound of γ^1, which means we consider the possibility that when the last subject is detected in our algorithm, the corresponding γ is lower than the c_L^1. This further compensates for the over-subtraction in our earlier iterations.

3.4 Localizing Multiple Moving Subjects When the Subject Count is Known

In this section, we discuss how we localize multiple moving subjects when the subject count is known. In SCPL, we track multiple subjects in parallel, unlike in the counting phase where we count the number of subjects sequentially. Radio interference is very complex and unpredictable, especially when multiple subjects are present and a link is affected by multiple people. In this case, it is hard to quantify the exact impact of a subject. Even after considering the cell link coefficient matrix $\mathcal{B}$, we may still overestimate (or, underestimate) a subject's impact on a link. These errors, while insignificant enough not to hurt the counting process, will lead to inferior localization results. On

Algorithm 2: Successive Cancellation-Based Device-free Passive Counting Algorithm

input : $\mathcal{D}$- The training data collected from L links among K cells
$\mathcal{S}$- The states $\{S_1, ..., S_K\}$ associated with the K cells
O- The testing data collected from L links when subjects are in unknown locations
$\mathcal{B}$ - The estimated location-link coefficient matrix generated from Algorithm 1
c_L^0, c_U^0- The lower and upper bounds of the 95% confidence interval when there is no subjects in the deployed area
c_L^1, c_U^1- The lower and upper bounds of the 95% confidence interval when there is one subject in the deployed area
output: C- The estimated number of subjects present in the deployed area

$C \leftarrow 0$
$\gamma \leftarrow \sum_{l=1}^{L} O^l$
// Presence detection
if $\gamma \leq c_U^0$ **then**
    return C;
// Count the present subjects
else
    while *true* **do**
        if $\gamma \geq c_U^1$ **then**
            // Estimate the most likely occupied cell
            $q \leftarrow \text{argmax}_{i \in \mathcal{S}} P(O|S_i)$
            // Remove the training data associated with the estimated cell in each round
            $\mathcal{D} \leftarrow \mathcal{D} \backslash \mathcal{D}_q$
            $\mathcal{S} \leftarrow \mathcal{S} \backslash q$
            // Update the testing data by removing the partial impact caused by the detected subject in each round
            for $l = 1 \rightarrow L$ **do**
                $O^l \leftarrow O^l - \beta_{ql} \mu_{ql}$
            $C \leftarrow C + 1$
            // Update the overall affect energy indicator
            $\gamma \leftarrow \sum_{l=1}^{L} O^l$
        else if $\gamma < c_U^1$ **then**
            $C \leftarrow C + 1$
            return C;

the other hand, parallel tracking keeps all the raw RSS values and can provide better results.

3.4.1 Understanding the Challenge of Localizing Multiple Subjects

Before presenting our localization algorithm, we first take a closer look at how multiple subjects collectively affect the RSS values and thus complicate the localization problem through empirical data. The complexity of this problem mainly stems from the multi-path effect [56], a typical error source in RF-based indoor localization. In this problem, multi-path can cause nonlinear interference in a radio space when multiple subjects are present. More precisely, when multiple subjects coexist in different locations, the resulting RSS value will not be simply the summation of the individual RSS values from a single subject independently in those locations. The gap between these two is larger when these subjects are close to each other. To validate this conjecture, we randomly select a few positions with certain distances apart. We first have one subject, A, collect the RSS measurements by standing stationary in these locations. Then, we involve another subject, B with similar height and weight as A, and have them stand in two different positions, say i and j. We use O_i and O_j to denote the measured RSS vector when A is standing in positions i and j independently, and O_{ij} the measured RSS vector when A and B are standing in positions i and j simultaneously. In a linear space, vector O_{ij} would be simply the summation of O_i and O_j. However, as mentioned before, this problem is nonlinear, especially when subjects are close to each other. To quantify the degree of nonlinearity, we define the *RSS Error Residual* as

$$\Delta O^l = O^l_{ij} - O^l_i - O^l_j,$$

for link l. A larger ΔO^l value indicates a higher non-linear degree. To articulate the nonlinearity nature, we remove link l if its O^l_{ij}, O^l_i, O^l_j values are all less than 1 because these links are actually not affected by the subjects in any case. We plot the histograms of the remaining O^l values in Figure 3.3.

From Figure 3.3, we have three main observations. Firstly, when the two subjects

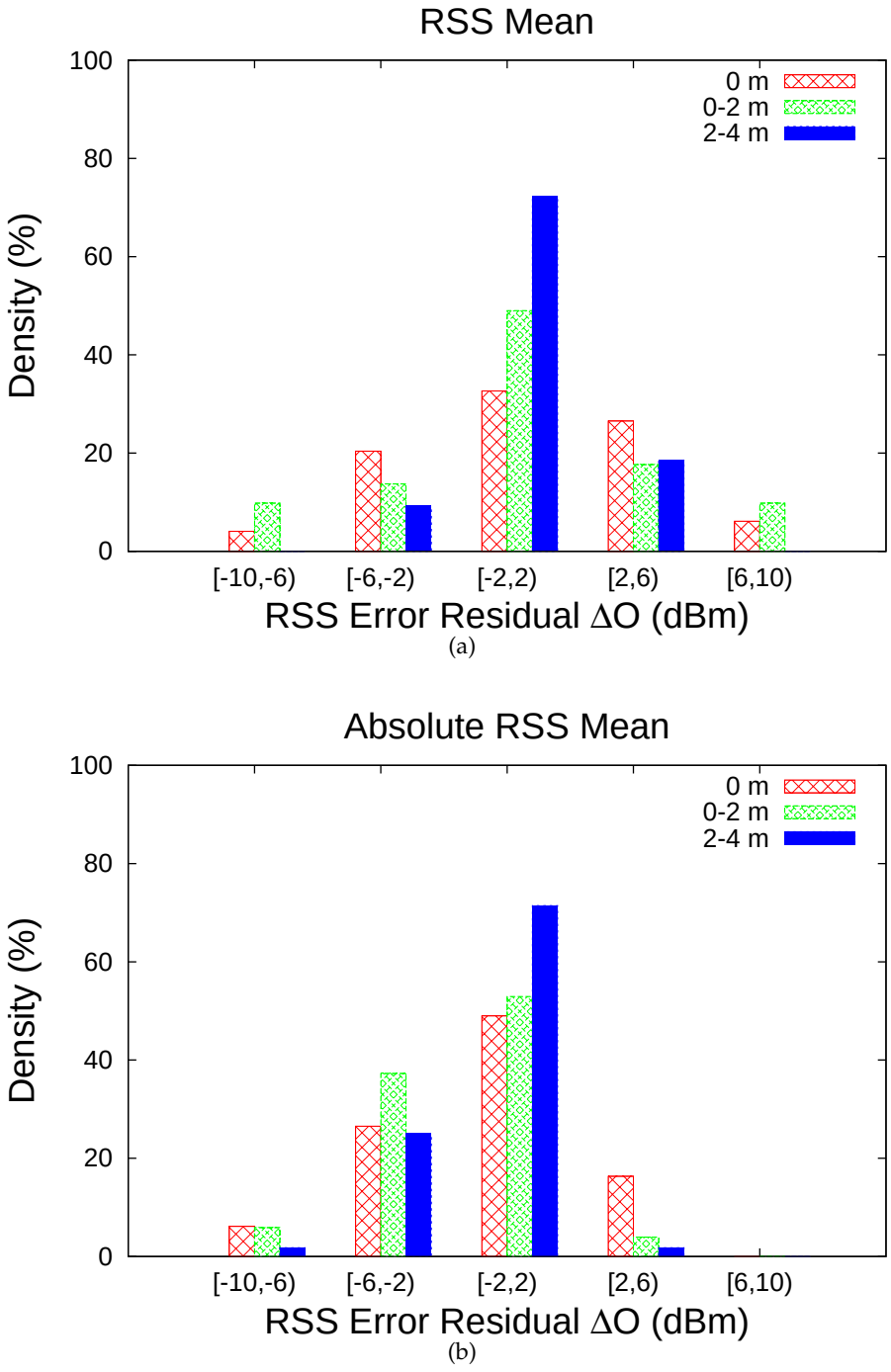


Figure 3.3: The RSS residual error forms a double-sided distribution when using RSS mean, while it is approximately single-sided distributed using absolute RSS mean.

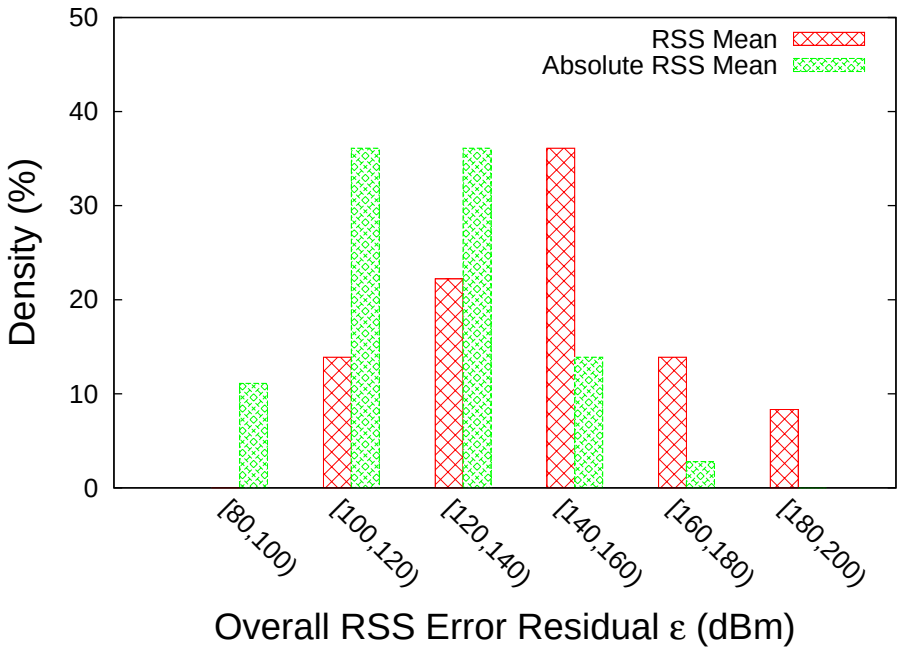


Figure 3.4: Absolute RSS mean has a smaller overall RSS error residual distribution.

stand side by side (i.e., the distance between them is 0 m), there are only about 30% and 50% chances that we see $|\Delta O^l| < 2$ for RSS mean and absolute RSS mean respectively, which validates our problem is indeed nonlinear. As the distance becomes longer than 2 m, the probability of having $|\Delta O^l| < 2$ rises to more than 70% for both RSS mean difference and absolute RSS mean difference. Secondly, the error residual can be negative under RSS mean difference, but is positive under absolute RSS mean difference in most cases, suggesting O_{ij} is consistently an underestimation of $O_i + O_j$. This property is desirable because it ensures Monotonicity.

Finally, we define the *total RSS Error Residual* as:

$$\varepsilon = \sum_{l=1}^{L} |\Delta O^l|,$$

which measures the deviation between the profiling data and the RSS measurement in a multi-subject problem. We plot the histogram in Figure 3.4 and observe that the absolute RSS mean has a smaller ε value, and thus more appropriate for our purposes.

3.4.2 Conditional Random Field Formulation

Tracking moving subjects actually introduces new optimization opportunities - we can improve our localization results by considering the fact that human locations from adjacent time intervals should form a continuous trajectory, which can be further modeled as a state transition process under conditional random field (CRF) [32]. CRFs are a type of discriminative undirected probabilistic graphical model. We use them to decode the sequential RSS observations into continuous mobility trajectories.

The first step towards formulating a conditional random field is to form the sensor model and transition model respectively. In our problem, we have K states: $\mathcal{S} = \{S_1, S_2, ..., S_K\}$. In a single-subject problem, state S_i means the subject is located in cell i. The sensor model essentially infers the current state based on the observation RSS vector O, which is to generate a cell likelihood map based upon O. For a single subject case, we would like to maximize the likelihood $P(q = S_i|O, \mathcal{D})$ when cell i is occupied. In other words, when the subject is located in cell i in the testing phase, we would like to maximize the probability that the estimated state/cell q matches the actually occupied cell i. We assume the observed RSS vectors in each state follow a multivariate Gaussian with shared covariance, as in [78], and denote

$$\delta_i(O) = P(O|S_i),$$

where

$$P(O|S_i) \sim \mathcal{N}(\mu_i, \Sigma).$$

However, the sensor model is imperfect because of the deep fading effect that can cause estimation error through only a few links[2]. Therefore, the cell associated with the maximum probability might be far from the ground truth.

Next, we look at the transition model. In each clock tick $t = 1, 2, ..., T$, the system makes a transition to state q_t. This process models the movement of a subject – the

[2]Because of deep fading from multipath, adjacent points can have dramatically different RSS values, leading to large estimation errors.

subject moves to a new cell in each tick. We choose a first order CRF, which means the next cell number depends on the current cell number, rather than any earlier history because we do not want to assume any specific human movement trajectories. In our model, subjects can either walk along a straight line, take turns or wander back and forth.

The subject's trajectory can thus be characterized as a parametric Markov random process with the *transition model* defined as the probability of a transition from state i at time $t-1$ to state j at time t in form of

$$T = P(q_t|q_{t-1}),$$

where

$$T_{ij} = P(q_t = S_j|q_{t-1} = S_i).$$

The intuition here is that people cannot walk through walls or cross rooms in a single tick. We believe these mobility constraints can be used to fix most of the errors in the sensor model caused by deep fades.

In our cell-based approach, we define the following:

Cell neighbors are a list of adjacent cells which can be entered from the current cell without violating mobility constraints.

Order of neighbor is defined as the number of cells a person must pass through to reach a specific cell from the current cell without violating mobility constraints. We assume the subject moves to a new cell every clock tick. For example, as far as cell i is concerned, the 1-order neighbors include its immediate adjacent cells, and its 2-order neighbors include the immediate adjacent cells of its 1-order neighbors (excluding i and i's first order neighbors).

Trajectory ring with radius r is defined as the area consisting of cell i, i's 1-order neighbors, 2-order neighbors, ..., up to its r-order neighbors. Particularly, 0-order trajectory ring consists of all the cells.

Let $\Omega_r(i)$ be the cells included in i's r-trajectory ring and let $N_r(i)$ be the size of

$\Omega_r(i)$. Our transition model thus becomes:

$$T_{ij} = \begin{cases} \frac{1}{N_r(i)} & \text{for} \quad j \in \Omega_r(i) \\ 0 & \text{for} \quad j \notin \Omega_r(i) \end{cases}$$

3.4.3 Localization Algorithm

Having constructed the sensor model and transition model, we can translate the problem of subject tracking to the problem of finding the most likely sequence of state transitions in a continuous time stream. The *Viterbi algorithm* [19] defines $V_j(t)$, the highest probability of a single path of length t which accounts for the first t observations and ends in state S_j:

$$V_j(t) = \underset{q_1,q_2,\cdots,q_{t-1}}{\operatorname{argmax}} P(q_1 q_2 ... q_t = j, O_1 O_2 ... O_t | T, \delta).$$

By induction

$$V_j(1) = \delta_j(O_1),$$

$$V_j(t+1) = \underset{i}{\operatorname{argmax}}\, V_i(t) T_{ij} \delta_j(O_{t+1}),$$

which is similar as discussed in [77].

Generalizing to the multi-subject case, we denote $\delta_{1:K}(O) = \{\delta_1(O), \delta_2(O), ..., \delta_K(O)\}$ from the sensor model to represent the likelihood of each state. We denote $Q = \{q^1, ..., q^C\}$, where C is the total number of present subjects. For the current state Q_t, we have $\binom{K}{C}$ possible permutations of subject locations. For each permutation j, we denote $Q_j = \{q^1, ..., q^C\}$ and compute the Viterbi score

$$F_j = \sum_{i=1}^{C} \delta_{q_t^i}(O_t) T_{q_{t-1}^i q_t^i}.$$

We then pick the j value that is associated with the maximum Viterbi score as the current state.

We describe our device-free multi-subject localization algorithm in Algorithm 3.

We believe we can achieve best localization results when we consider 1 or 2-order trajectory ring, which is better than the 0-order case used in our earlier work [78], and is also confirmed by our experimental results presented in Section 3.6.

Algorithm 3: Trajectory-Based Device-free Multi-subject Localization Algorithm

input : $\mathcal{D}$- The training data collected from L links among K cells
T- The transition model
$O_{1:t}$- The testing data collected from L links when subjects are in unknown locations
C- The estimated number of present subjects in the deployed area
Q_1- The initial state(s) of the present subjects
output: $Q_{1:t}$- The most like sequence of the trajectories of the present subjects

1 **for** $i = 2 \rightarrow t$ **do**
2 $\quad \delta_{1:K}(O_i) \leftarrow P(O_i|\mathcal{D})$
3 $\quad \Pi \leftarrow$ is the set of all the possible permutations of $\binom{K}{C}$
4 $\quad Q_i \leftarrow \text{argmax}_{j \in \Pi}$ `ViterbiScore`$(Q_{i-1}, Q_j, \delta_{1:K}(O_i), T)$

3.5 Experimental Setup

In this section, we briefly describe the experimental setup, the data collection process and the metric we use for performance evaluation.

3.5.1 System Description

The radio devices used in our experiments contain a Chipcon CC1100 radio transceiver and a 16-bit Silicon Laboratories C8051-F321 microprocessor powered by a 20 mm diameter lithium coin cell battery, the CR2032. The receivers have a USB connector for loss-free data collection but are otherwise identical to the transmitters. In our experiments, the radio operates in the unlicensed bands at 909.1 MHz. Transmitters use MSK modulation, a 250 Kbps data rate, and a programmed output power of 0 dBm. Each transmitter periodically broadcasts a 10-byte packet (8 bytes of sync and preamble and 2 bytes of payload consisting of transmitter's id and sequence number) every 100 millisecond. When the receiver receives a packet, it measures the RSS values and

wraps the transmitter id, receiver id, RSS, timestamp (on the receiver side) into a "data packet". The packets are forwarded to a centralized system where the data can be analyzed by independent "solvers" that perform various data processing functions. These include packet loss calculations [18], mobility detection [29], counting, localization, and data interpolation. More detail of the system can be found on the Owl Platform website [1].

3.5.2 Data Collection

In our experiments, the RSS data is collected as a mean value over a 1 second window for each link. We choose a 1 second window because a normal person can at most walk across one cell during a second. In the training phase, a single subject made random walk for 30 seconds in each cell and collected 30 RSS vectors as the profiling data. In this testing phase, we designed four scenarios for each environment, and in each scenario the subject(s) individually form a continuous mobility trajectory for about 30 seconds. The subjects are walking at a speed of about 0.5 m per second. The training phase was performed in the early morning while the testing phase happened the afternoon of the same day.

3.5.3 Deployment Cost

In this study, we deployed our system in two different indoor settings which we will shown in Section 3.6. Our "solver" is running on a laptop (Intel i7-640LM 2.13GHz, 8GB RAM). For the 150 m^2 setting, it took 15 minutes to collect the training data, 0.003 seconds for the solver to fit the model parameters, and 3.4 seconds to compute the location-link correlation coefficients. The second area was 2.7 times larger (400 m^2), but data collection only took 30 minutes, the solver was actually faster (0.002 seconds), and the time to compute the correlation coefficients only increased by a factor of about 1.5 (5.3 seconds).

3.5.4 Performance Metrics

We use the following performance metrics to measure our counting and localizing algorithms.

Counting Percentage is given by:

$$1 - \frac{|\hat{C} - C|}{C},$$

where $\hat{C}$ is the estimated subject count and C is the actual subject count.

Error Distance is defined as:

$$d(Q, \hat{Q}) = \frac{1}{C} \min_{\pi \in \Pi} \sum_{i=1}^{C} d(q^i, \hat{q}^{\pi(i)}),$$

where Π includes all the possible permutations of $\{1, 2, ..., C\}$, $d(q, \hat{q})$ is the Euclidean distance between the ground truth q and the estimated position $\hat{q}$. $Q = \{q^1, q^2, ..., q^C\}$ and $\hat{Q} = \{\hat{q}^1, \hat{q}^2, ..., \hat{q}^C\}$ are within the pre-profiled finite states $\mathcal{S} = \{S_1, S_2, ..., S_K\}$. In this study, q is the subject's actual location and $\hat{q}$ is her estimated location (i.e., center of the estimated cell).

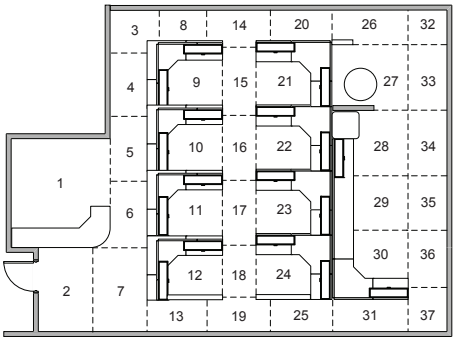
3.6 Experimental Results

In this section, we summarize the results we have obtained from two indoor settings. In each setting, we had multiple subjects each walking along a trajectory.

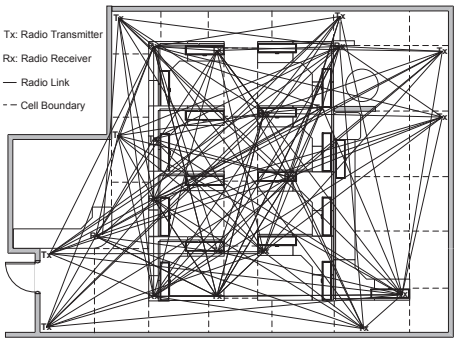
3.6.1 Results from Office Setting

Our first setting is a typical office environment, consisting of cubicles and aisles with a total area of 150 m^2. The environment is quite cluttered as shown in Figure 3.5(a). The area is broken down to 37 cells such as cubicles and aisle segments, as shown in Figure 3.5(b). We utilized 13 radio transmitters and 9 radio receivers, whose locations and corresponding link LoS's are shown in Figure 3.5(c). Here, we need to point out that these devices were installed for some earlier projects, not specifically for this one,

(a) Test Field



(b) Cell Locations



(c) Radio Link Distribution

Figure 3.5: In (a), we show the office in which we deployed our system. In (b), we show that the office deployment region is partitioned into 37 cubicle-sized cells of interest. In (c), we show the locations of the pre-installed 13 radio transmitters, 9 radio receivers and the corresponding Line-of-Sight links.

and therefore, the link density per cell is non-uniform. This, however, represents a more realistic setting, through which we can show that SCPL can achieve good results without dedicated sensor deployment.

We had four subjects (A, B, C and D) in this series of experiments. We went through several example scenarios and illustrate them in Figure 3.6:

- *One Subject Scenario:* A left her boss's office, and walked along the aisle to her cubicle.
- *Two Subject Scenario:* When B entered the room, A was walking on the aisle towards him. B waited until they met and walked together for some time, and then separated to go back to their own seats.

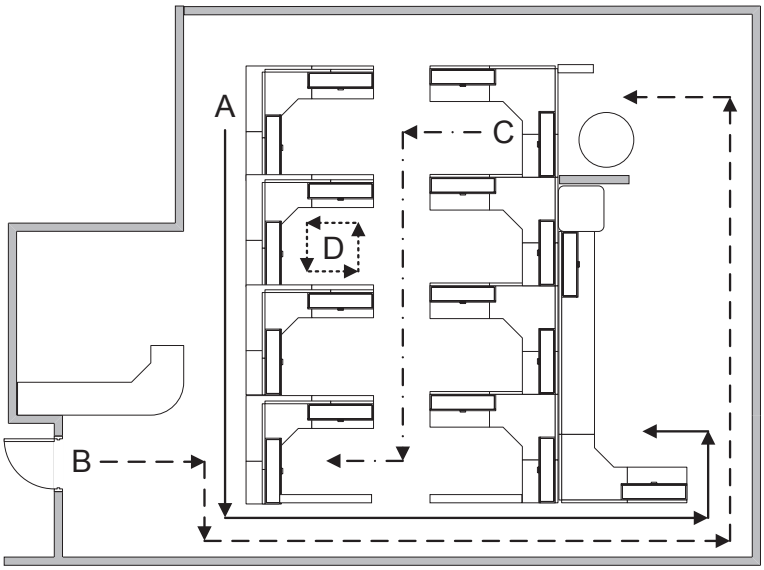


Figure 3.6: We show the experimental trajectories of subjects A, B, C and D in the office setting. Note the the trajectories of A and B are partially overlapped at the same time.

- *Three Subject Scenario:* While A and B followed the movement patterns in the above two subject scenario, C walked on the other aisle from one cubicle to another.
- *Four Subject Scenario:* While A, B, and C followed the movement patterns in the above three subject scenario, D was sitting on her seat.

Counting Results

The difficulty of subject counting increases when multiple subjects walk together (in the same cell). Thus, we present our counting results in the following three ways: (a) all the experimental data (referred to as *mixed*), (b) the experimental data for when multiple subjects walked together and thus had overlapping trajectories (referred to as *overlap trajectory*), and (c) the experimental data for when multiple subject trajectories did not overlap (referred to as *non-overlap trajectory*). Figure 3.7 shows the counting percentages in all three cases.

We observe that when we have multiple subjects, the counting percentage is higher

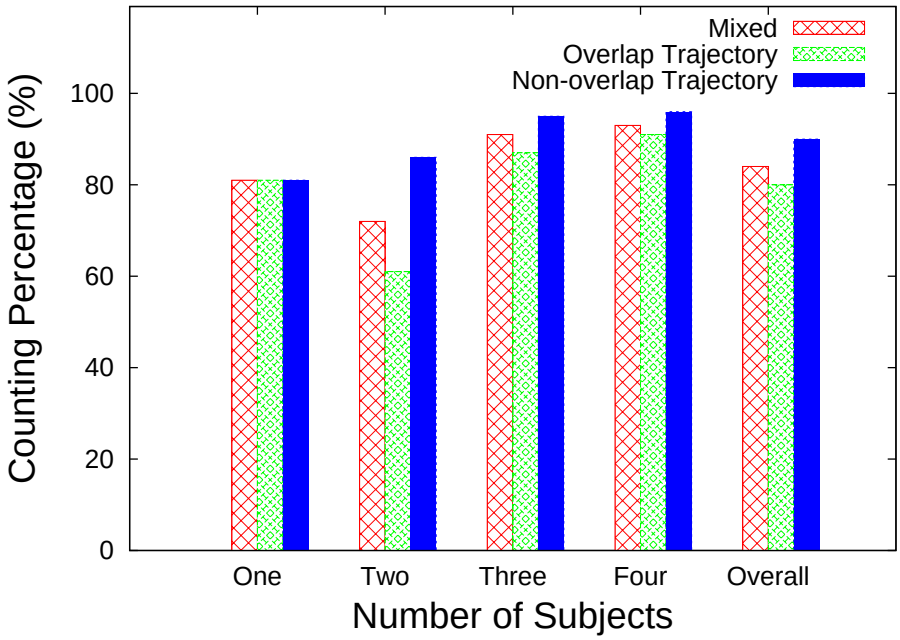


Figure 3.7: In a multi-subject case, our counting algorithm has a better performance when their trajectories are not overlapped than overlapped.

in the non-overlap trajectory case. The average counting percentage across all cases is 84%, the average counting percentage for non-overlap cases is 90%, and the average counting percentage for overlap cases is 80%.

Next, we show the performance improvement of subtracting a normalized RSS contribution by location-link coefficients compared to directly subtracting a cell's mean RSS change. We show the counting percentage results in these two cases in Figure 3.8. When we have one or two subjects, the non-linearity is not very obvious, and these two methods have very similar results. When we have more than two subjects, the non-linearity of the signal change becomes very pronounced, and using a normalized RSS contribution can yield better counting results. Specifically, we observe a 36% improvement with three subjects, and a 24% improvement with four subjects.

Finally, we show our subject counting results in Figure 3.9, in which all the four tests last 32 seconds. In the single-subject case, we see two individuals, between time tick 12 and 20. This is likely because there is an overestimate of γ near cells 13, 19, and 25, because of a denser than average link space or proximity to the receiver.

In the two-subject case, we under-estimate the subject count by one between time

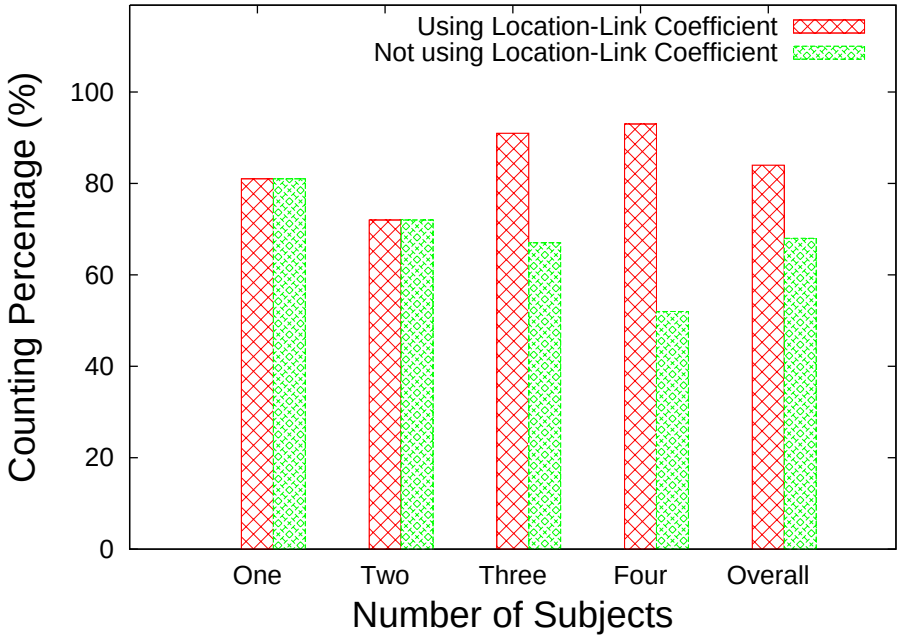


Figure 3.8: Counting percentage improvement when the RSS change is normalized by location-link coefficients in the office setting.

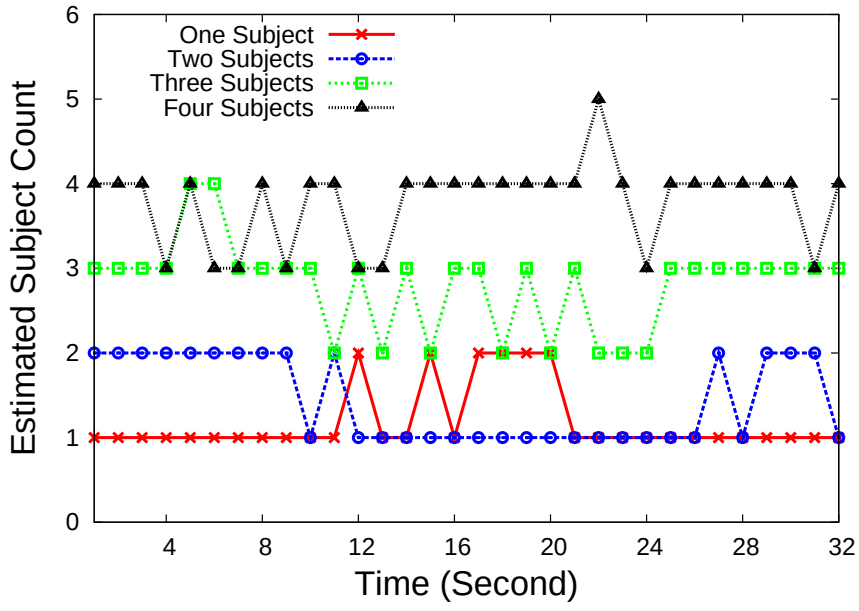


Figure 3.9: Estimated subject count over time using our successive cancellation-based counting algorithm in the office setting.

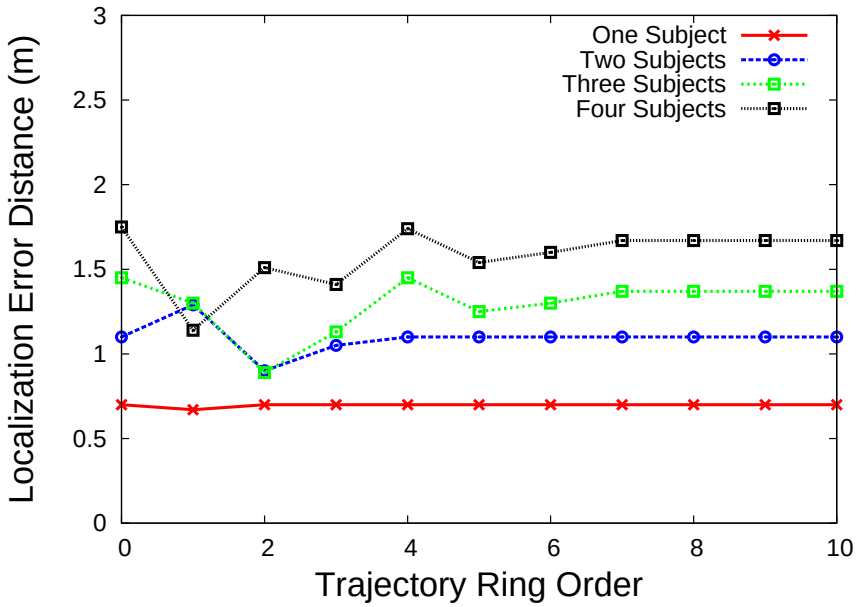


Figure 3.10: We achieve best localization accuracy averaging all the test cases when we adopt 1 or 2-order trajectory rings in the office setting.

tick 10 and 26 because the two subjects merged their trajectories in those time periods. The errors caused by temporally overlapping trajectories can also be easily addressed as follows. We continuously run the counting algorithm, and once we notice the estimated subject count suddenly drops, we check their locations before the sudden drop. If no subject's location was close to the exit, then we can conclude that two or more (depending upon the change in the count) were in close proximity. Of course, this information should be validated from the subject location information. For the three subjects case, we see the same problem when subjects A and B merge their trajectories. For the four subject case, this error is reduced a bit because subject D is always in cell 10, where has a relatively high density of radio links.

Localization Results

We show the mean of localization error distances in Figure 3.10 with different ring order parameters. In our setting, we choose 10 as the upper bound of the ring order because all cells are within 10 hops of each other.

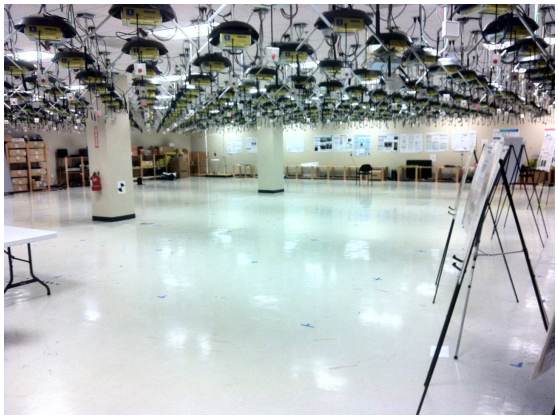
Our first observation is that the use of the trajectory information can improve the

localization performance by 13.6% – the overall mean localization error distance drops from 1.25m (with 0-order trajectory ring) [78] to about 1.08m (with 1-order trajectory ring). We note that the error distance for a single subject does not benefit from using trajectory information because the profiling data is good enough for this case [83, 78]. Multiple subjects, especially when they are close to each other, will cause non-linear radio interference, and thus the data collected from the mutually affected links alone cannot give very accurate localization results. Therefore, the sensor model alone is insufficient for high accuracies. Secondly, we observe that the localization results are less accurate in those cells with lower radio link densities, such as in cell 34-37, because subjects may cause negligible changes to the RSS space at a few points in those cells. Thirdly, trajectory information helps prevent the error distance increases dramatically as the increasing number of subjects. Finally, our environment is an office space consisting of cubicles and aisles, and the possible paths a subject can take are rather limited. As a result, we achieve the best localization accuracies with 1 or 2 order trajectory ring. Due to the movement constraints, a higher order trajectory ring has the same result as not considering any neighbors at all (i.e., 0 ring order). We hypothesize that this may not be true in a more open indoor environment such as (large) homes, malls and museums.

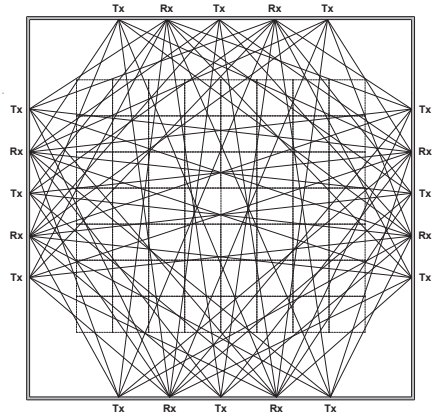
3.6.2 Results from Open Floor Space

The second test setting is a more open floor of total 400 m^2, as shown in Figure 3.11(a). We used this setting to model an open hall with a few posters on exhibition, and SCPL can be used to detect traffic flow and infer the most popular poster. We deployed 12 transmitters and 8 receivers in such a way that the link density has a relatively even distribution across the cells, as shown in Figure 3.11(b). We would like to point out that we used fewer devices in this setting than in the previous one, though this one had a larger area. Also, this environment is even more challenging in that half of the radio devices are deployed on a wall which also has dozens of computers and other metal parts, significantly degrading radio propagation.

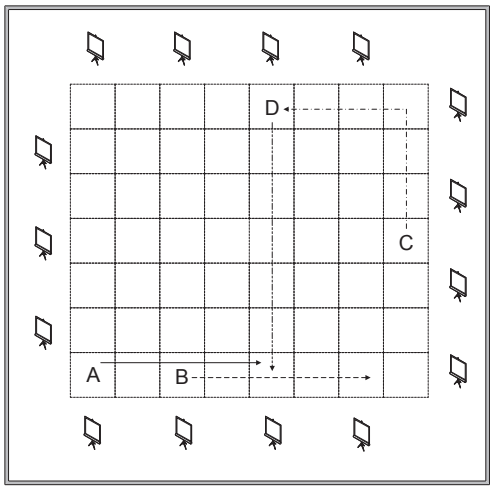
The space was partitioned into a uniform grid of 56 cells, and we involved four



(a) Test Field



(b) Radio Link Distribution



(c) Test Trajectories

Figure 3.11: In (a), we show the open floor space used for poster exhibition in which we deployed our system. In (b), we show the locations of the 12 radio transmitters, 8 radio receivers and the corresponding Line-of-Sight links. In (c), we show the experimental trajectories of subjects A, B, C and D in the open floor space which is partitioned into a uniform grid of 56 cells.

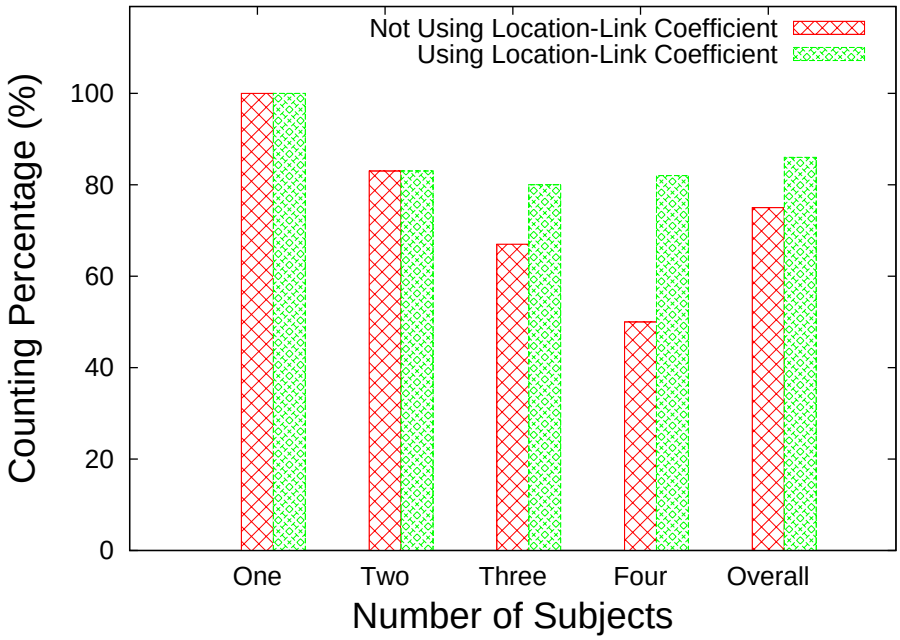


Figure 3.12: Counting percentage improvement when the RSS change is normalized by location-link coefficients in the open floor space.

different subjects in this test and show their trajectories in Figure 3.11(c). We repeated the same four scenarios as in the previous setting. We plot our counting results in Figure 3.12. We achieve a 100% counting percentage when there was only a single subject, which is better than the previous setting because the link density is more even in this case.

We achieve a counting percentage of 83%, 80%, and 82% for two, three and four subjects respectively, resulting in a 86% counting percentage in total. We have achieved better results when we normalize a subject's impact from a certain cell on the RSS with the location-link coefficients. We observe similar trends as in the previous setting: the results are the same for one or two subjects, and improved from 67% to 80%, and from 75% to 86% when we have three and four subjects, respectively. The estimated subject is shown in Figure 3.13.

We present the localization results in Figure 3.14. In the localization part, we observe similar patterns as in the previous setting: we achieve better localization accuracy using trajectory information. We achieve the best localization accuracy when we adopt the 2 order trajectory ring, which is 1.49 m, a 35% improved compared to the

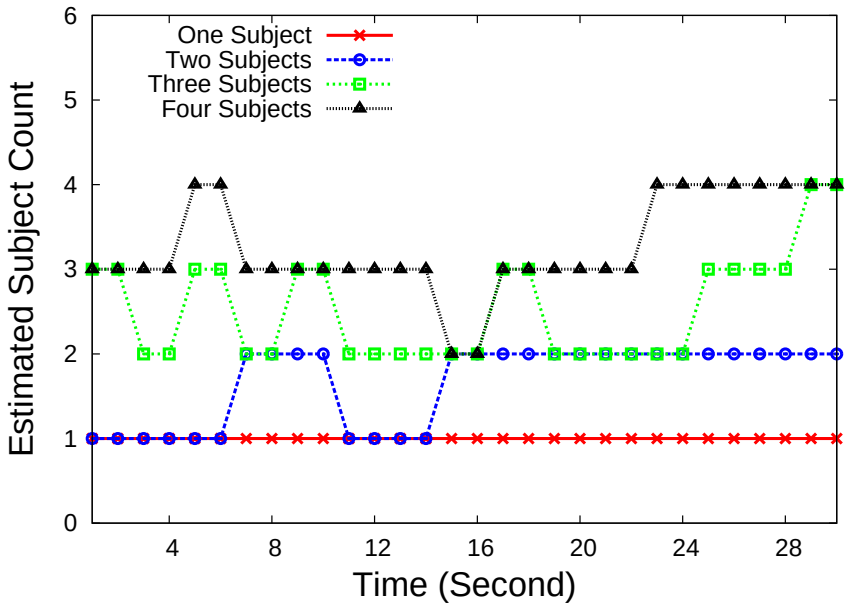


Figure 3.13: Estimated subject count over time using our successive cancellation-based counting algorithm in the open floor space.

0-order trajectory ring case [78].

3.7 Limitations and Future Work

In this section, we discuss the limitation of SCPL.

3.7.1 Algorithms

Recognizing human mobility constraints in indoor environments leads to different trajectory-based tracking optimizations. Under our framework of discretized physical space, our localization algorithm relies on a greedy search for the optimal solution to find the most likely trajectories followed by the individuals. Unfortunately, this has factorial computation complexity because it involves C-permutations of K[3] and potentially introduces prohibitive overhead to meet real-time requirements, especially when K grows rapidly in a large-scale environment. However, as we observed from the experimental results from the two different settings, we have achieved the best localization accuracies using only the 1 or 2-order trajectory ring, which means we can

[3] C is the subject count and K is the total number of cells.

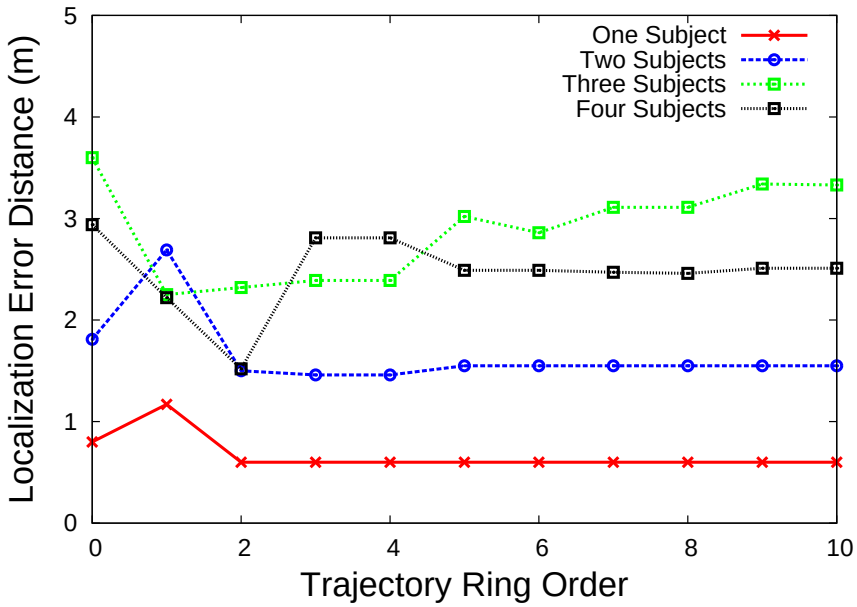


Figure 3.14: We consistently achieve best localization accuracy when we adopt 1 or 2-order trajectory rings in the open floor space

not only achieve good accuracy, but also significantly reduce the computational complexity by reducing the permutation space from $\binom{K}{C}$ to $\binom{K'}{C}$, where K' is the cell union of each individual's 1 or 2-order trajectory rings. Under 1-order trajectory ring, it took 0.87 seconds and 0.88 seconds to count and localize four subjects in our two different settings respectively. We would expect that it will take more than 1 second to track at least five subjects, which fails to afford real-time tracking requirement with this hardware. Another family of trajectory based tracking incorporates a particle filter [58], such as the one used in [73, 45]. However, the primary weakness of particle filters is the computational complexity required to run the algorithm for the large number of particles needed to achieve accurate results. For example, 500 particles were needed for tracking each individual and it took 7.6 seconds for four subjects in each time step, as reported in [45]. Overall, there is plenty space to optimize the trade-off between accuracy and computational cost in tracking multiple subjects for future work.

3.7.2 Long-term Test

In a long-run test, any RF-based localization schemes suffer not only from temporal fading, but also from environmental changes. A small piece of metal can change the tuning of the antenna shift the radiation pattern or even the radio frequency of the nearby transmitter or receiver. Either or both of these effects can change the underlying propagation pattern and, hence, the RSS values on the links. To avoid frequent manual recalibration, we present two schemes in our earlier work [78, 79] to maintain the localization accuracy over a long-term test. In [78], we simply remove the radio links experiencing deep fading by watching RSS values over time, which is able to maintain a cell estimation accuracy of 90% over one month. In [79], we present a camera-assisted auto recalibration – when the camera occasionally turns on, it localizes the subject and calibrates the RF data automatically. Both schemes have limitations: the performance of the first scheme will degrade when the number of remaining links is too small, while the second one needs extra hardware. Realizing these limitations, we will investigate sophisticated auto-calibration methods as part of the future work.

3.8 Related Work

In this section, we briefly review the related literature in RF-based counting and localizing device-free human subjects.

3.8.1 Device-Free Counting

Nakatsuka et al. [44] first demonstrated the feasibility of using radio signal strength to estimate the crowd density. The authors setup two radio nodes and observe that RSS decreases as the number of subjects increases when they are all sitting between the nodes. We, however, point out that SCPL is the first work that systematically counts device-free subjects in large scale deployment, to our best knowledge.

	Grid Array [84]	RTI [25]	NUZZER [62]	SCPL
Meausred physical quantity	RSS variance	RSS attenuation	RSS change	RSS change
Non-LoS localization	No	Yes	Yes	Yes
Nodes density	High	High	Low	Median
Prior knowledge of node locations	Yes	Yes	No	No
Tracking static subjects	No	Yes	Yes	Yes
Deployment scale	Median	Small	Large	Large
Training overhead	Low	Low	High	Median

Table 3.1: Comparison of different RF-based passive localization systems.

3.8.2 Device-Free Localization

In 2006, Woyach et al. [74] first experimentally demonstrated the feasibility of localizing device-free subjects by observing a difference in RSS changes by a subject moving between (resulting signal shadowing effect) and in the vicinity (causing small-scale fading) of a pair of transmitter and receiver. From then on, several DfP approaches have been proposed in the literature, which can be broadly categorized into two groups as follows.

Location-based schemes: This approach is also known as "fingerprinting", a popular approach for RF-based localization. It was first studied in [83] in the context of passive localization. The authors first collect a radio map with the subject present in a few predetermined locations, and then map the test location to one of these trained locations based upon observed radio signals. This method explicitly measures the multipath effect on RSS in each different position, and thus avoids modeling errors. In addition, it does not require a node deployment as dense as in link-based schemes because when the subject is in the position has no intersection with any radio LoS links, the RSS ground truth still can provide a distinguishable record from other positions. This work is extended to a much larger deployment in Nuzzer [62]. In [78], Xu et al. propose to formulate this localization problem into a probabilistic classification problem and use a cell-based calibration with random walk method profiling the system in order to mitigate the error caused by the multipath effect in cluttered indoor environments, improve the localization accuracy and meanwhile reduce the profiling overhead. However, the downside of fingerprinting is also evident: the calibration procedure is relatively tedious.

Link-based schemes: These techniques look for those radio links close to the target subjects and further determine the locations of the targets based on the RSS dynamics. Zhang et al. [85] set up a sensor grid array on the ceiling to track subjects on the ground. An "influential" link is one whose RSS variance exceeds a empirical threshold. The authors determine a subject's location based upon the observation that these influential links tend to cluster around the subject. This technique forms a consistent link-based model to relate the subject's location relative to the radio link locations. In [86]. the authors extend their algorithms to track up to subjects separated by at least 5 m. In [84], the monitored area is partitioned into different triangle sections, and the nodes in neighbor section are working at different communication channels to reduce the interference among nodes. The authors applied support vector regression model to track up to two subjects. The fundamental limitations of this series of work is that (i) not all the monitored places have the facilities to mount nodes on the ceiling; (ii) this work uses RSS variance as the data primitive, which is essentially the amplitude and phase shift of the ground reflection multipath caused by the of human subjects only in motion. In other words, the system might fail to work if the subjects stop walking. Another sets of work following Link-based DfP is radio tomographic imaging (RTI). Wilson et al. [71] use tomographic reconstruction to estimate an image of human presence in the deployment area of the network. RSS attenuation is used as data primitive in [71], which effectively works in outdoor or uncluttered indoor space without rich multipath. Recognizing the nature of multipath fading, Wilson et al. defined the concept of fade-level [73], which captures the ambient RSS characteristics of each link and categorize the links into deep fade (the RSS will increase on average when the LoS is blocked) and anti-fade (the RSS decreases when the LoS is obstructed) through fitting the calibration data to a skewed Laplace distribution. The authors demonstrate this technique's effectiveness through testing in same setting over time and a totally different setting without the effort of re-estimating the model parameters. Kaltiokallio et al. [25] further exploit channel diversities to enhance the tracking accuracy. Taking the framework of RTI, another sets of work is done based on sequential Monte Carlo sampling techniques. Chen et al. [11] propose to use auxiliary particle filtering method to

simultaneously localize the nodes and a single subject in an outdoor setting. In [66], the author introduce a measurement model which assumes the attenuation in RSS due to the simultaneous presence of multiple subjects on the LoS is approximately equal to the sum of the attenuations caused by the individuals. This model is then applied in [45, 46] for tracking up to four subjects in outdoor and indoor settings. In general, link-based schemes have two advantages: (i) the algorithms are robust to the environmental change because the subject's location is directly estimated based on its relative distance to each individual radio link LoS; (ii) it requires less calibration effort - only sensor locations and ambient RSS for each link is needed. However, it requires a dense nodes deployment to provide enough radio LoS links to cover all the physical space.

Finally, we summarize the differences between our system and the recent DfP RF-based localization systems in Table 3.1.

3.9 Conclusion

In this chapter, we present SCPL, an accurate counting and localization system for device-free subjects. We demonstrate the feasibility of using the profiling data collected with only a single subject present to count and localize multiple subjects in the same environment with no extra hardware or data collection. Through extensive experimental results, we show that SCPL works well in two different typical indoor environments of 150 m^2 (office cubicles) and 400 m^2 (open floor plan) deployed using an infrastructure of only 20 to 22 devices. In both spaces, we can achieve about an 86% average counting percentage and 1.3 m average localization error distance for up to 4 subjects. Finally, we shows that though a complex environment like the office cubicles is expected to have worse radio propagation, we can leverage the increased mobility constraints that go with a complex environment to maintain or even improve accuracy in these situations.

Finally, we point out that if we rely on a single subject's training data, the number of subjects that can be accurately counted and localized is rather limited. We had success with up to 4 subjects, but were not very successful with more subjects. In our

future work, we will look at how we can accurately localize a larger number of subjects with reasonable overheads.

Chapter 4

Crowd++: Unsupervised Speaker Counting with Smartphones

In this chapter, we present SCPL, a framework aiming to count and localize multiple people using radio signal strength at one time.

4.1 Introduction

The most direct form of social interaction occurs through the spoken language and conversations. Given its importance, for decades scientists have proposed diverse methodologies to analyze the audio recorded during people's conversations to distill the various attributes that characterize this particular social interaction. In addition to the most obvious attributes of a conversation, i.e., its content [54], several types of contextual cues have also received attention including speaker identification, conversation turn-taking, and characterization of a social setting [14, 38, 24]. We, however, note that one of the most important contextual attributes of a conversation, namely, speaker count, has been largely overlooked. Speaker count specifies the number of people that participate in a conversation, which is one of the primary metrics to evaluate a social setting: how crowded is a restaurant, how interactive is a lecture, or how socially active is a person [53, 41]. In this chapter, we aim to accurately extract this attribute from recorded audio data directly on off-the-shelf smartphones, without any supervision, and in different use cases.

Most of the previous studies that focused on the extraction of conversation features all share a common thread: they often require specialized hardware – such as microphone arrays, external dongles pairing with mobile phones, or video cameras – and complex machine learning algorithms built upon supervised training techniques

requiring the collection of large and diverse data sets to bootstrap the classification models. The support of powerful backend servers is also often needed to drive these algorithms.

Given that smartphones are becoming increasingly powerful and ubiquitous, it is natural to envision new social monitoring architectures, with the smartphones being the only sensing and computing platform. In pursuit of these goals, we design a system called *Crowd++*, where we exploit the audio from the smartphone's microphone to draw the social fingerprints of a place, an event, or a person. We do so by inferring the number of people in a conversation – but not their identity – as well as their interactions from the analysis of the voices contained in the audio captured by the smartphones, *without any prior knowledge of the speakers and their speech characteristics*. Audio inference from smartphones' microphones has been previously used to characterize places and events by picking up different sound cues in the environment [4]. However, for the first time, we show how to infer the number of speakers in a conversation through voice analysis using the audio recorded on off-the-shelf smartphones.

Crowd++ is unique given its number of contributions: (i) it is entirely distributed, with no infrastructure support; (ii) it applies completely unsupervised learning techniques and no prior training is needed for the system to operate; (iii) it is self-contained, in that, the sensing and machine learning computation takes place entirely and efficiently on the smartphone itself as shown by our implementation on four different Android smartphones and two tablet computers; (iv) it is accurate, as shown by experiments where Crowd++ is used in challenging environments with different audio characteristics – from quiet to noisy and loud – with the phone both inside and outside a pocket, and very short audio recordings; and (v) it's energy and resource-efficient.

In spite of Crowd++ not being perfect and potentially affected by limitations – the count is based on active speakers and noise can possibly impact the count accuracy – we still believe that ours is a competitive approach in many different application scenarios. In the social realm for example: people are often interested in finding "social hotspots," where occupants engage in different social behaviors: examples are restaurants, bars, malls, and meeting rooms. What if we could know in advance the

number of people in a certain bar or restaurant? It might help us make more informed decisions as to which place to go.

While Crowd++ may be deemed only as an initial step, we show that faithful people count estimates in conversations can nevertheless be achieved with sufficient accuracy. We implement Crowd++ on four Android smartphones and two tablet computers and collect over 1200 minutes of audio over the course of three months from 120 different people. The audio is recorded by Crowd++ in a range of different environments, from quiet ones – home and office – to noisy places like restaurants, malls, and public squares. We show that the average difference between the actual number of speakers and the inferred count with Crowd++ is slightly over 1 for quiet environments, while being no larger than 2 in very noisy outdoor environments. We conjecture that this accuracy is adequate and meaningful for many applications – such as social sensing applications, crowd monitoring and social hotspots characterization just to name a few – that don't necessitate exact figures but only accurate estimates.

4.2 Motivation and Challenges

Speaker count is an important type of contextual information about conversations. Crowd++ is able to infer the number of speakers in a dialog without requiring any prior knowledge of the speech characteristics of the involved people because of its unsupervised nature. We believe that the ability to capture this information can support different classes of applications, some of which are summarized below.

Crowd Estimation and Social Hotspots Discovery. With Crowd++ it would be possible to estimate the number of people talking in certain places, such as restaurants, pubs, malls, or even corporate meeting rooms. This information is useful to assess the occupancy status of these places.

One question that comes to mind is: Why do we need a solution like Crowd++ to infer the number of people in a place? Wouldn't be enough to simply count the number of WiFi devices associated with an access point, piggyback to a bluetooth scan result, measure co-location, use computer vision techniques to analyze the number

of people in video images, or even use active methods that require the transmission and analysis of audio tones? The answers to these questions are quite straightforward: none of these techniques in isolation is the solution to the problem. In order to read the association table of an access point there is a need to have access to the WiFi infrastructure, which is often not allowed. Even if possible, a person with several WiFi devices may generate false positives. A count based on the result of a bluetooth discovery [70] is error-prone because of the likelihood of reaching out to distant devices. RF-based device-free localization techniques [76] require the support of an infrastructure of several radio devices. Acoustic-based counting engines as in [26] are error-prone because of surrounding noise and audio sensitivity to clothes. Counting people through computer vision techniques [10] requires customized infrastructure, suffers from privacy concerns, and is limited by lighting condition. Crowd++ inference is instead based on a much more localized event – speech – that can significantly scope the count inference to specific geographic regions. It's also passive, since no active sounds by the devices need to be played.

We generally assume that people usually engage in conversations in social public spaces such as restaurants, bars, or conference rooms. We also acknowledge that in other places, such as subway stations or movie theaters, silence is predominant, making it difficult for Crowd++ to properly operate. We, however, note that Crowd++ should not be deemed as a replacement of any of the existing approaches. Rather, it should be seen as a complementary solution that can be useful to boost the crowd count accuracy by working in concert with different techniques. Prior information about a certain place – such as the average number of people attending the place – combined with the properties of statistical sub-sampling can also be used to boost the final count accuracy.

Personal Social Diary. Doctors analyze their patients' social patterns to predict depression or social isolation and take early actions. Rather than using ad-hoc hardware as in [53], which could potentially perturb the quality of the measurements, Crowd++ is installed on the smartphones of people potentially affected by depression and operates unobtrusive monitoring in a much more scalable, and less invasive fashion. These

patients' social pattern could in fact be drawn by the social engagement captured by Crowd++ as the patients go about their daily lives.

Participant Engagement Estimation. What if a teacher could assess, after a lecture, the level of engagement of their students by simply looking at the number of students participating in discussions during the lecture and the frequency of the discussions? This could be used as an indirect measure of the class engagement and of the teacher's effort in improving the quality of their teaching. Students would in turn be motivated to run Crowd++ on their devices in order to share with their friends, and in turn apprehend from other students, information on the most lively lecture on campus.

4.2.1 Challenges

As in other smartphone audio inference applications, Crowd++ is affected by some challenges: the phone's location, e.g., in or out of a pocket or bag, smartphone's hardware constraints, and noise polluting the audio are the main limiting factors. Despite these limitations, we show through the development and evaluation of Crowd++ that the system is able to efficiently and accurately perform speaker count in a diverse set of environments and settings.

4.3 Privacy

It is quite natural to raise privacy concerns when doing audio analysis. These concerns become more serious when the audio is captured with a smartphone, which is always with the user, even in private spaces. With this in mind, we take specific steps to make sure that users' privacy is preserved.

Speakers' identity is never revealed. Crowd++ isn't able to associate a voice fingerprint to a specific person and it's designed to only infer the number of different speakers in an anonymized manner. Crowd++ could potentially identify only the phone's owner if the algorithm was actively trained to recognize the owner's voice. Identification of the owner may be optionally added to either improve the speaker count accuracy or in personal social diary applications.

The audio analysis is always performed locally on devices in order to avoid sensitive data leaks. The audio is deleted right after the audio features computation. Should communication with backend be needed, the servers should be trusted and off-the-shelf encryption methods for the communications should be put in place. Only features extracted from the audio, rather than the raw audio itself, should be sent to the server.

To guarantee the user's privacy when the data is sent to a backend server and to prevent attacks that exploit the audio features to reconstruct the original audio, measures such as the ones proposed by Liu et al. [34] should be put in place. In this work, it is shown how to manipulate the features to a point that they are still effective for a machine learning algorithm to infer events while, however, obfuscating the underlying content of the raw audio.

Finally, by giving users the ability to configure the application's settings, Crowd++ should be allowed to work only in specific locations – say, in public places. Through geo-fencing technologies, the application could be automatically activated and deactivated as directed by the user's pre-selected policies: e.g., activate it in the office and in restaurants but not at home.

4.4 System Design

Crowd++ estimates the number of active speakers in a group of people. It consists of three steps: (1) *speech detection*, (2) *feature extraction*, and (3) *counting*. In the speech detection phase, we extract the speech segments from the audio data by filtering out silence periods and background noise. In the feature extraction phase, we compute the feature vectors from the active speech data. In the counting phase, we first select the distance function that is used to maximize the dissimilarity between different speakers' voice, and then apply an unsupervised learning technique that, operating on the feature vectors with the support of the distance function, determines the speaker count. An overview of the Crowd++ pipelined approach is shown in Figure 4.1.

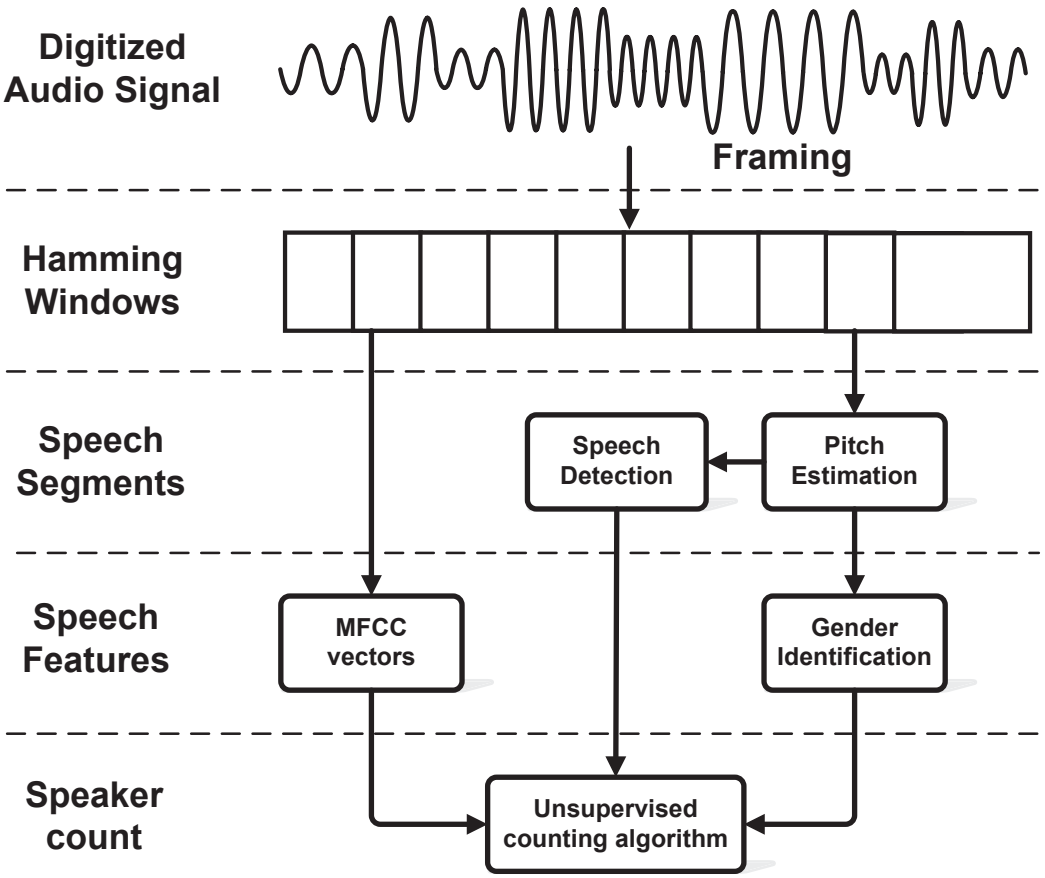


Figure 4.1: Crowd++ sequence of operations.

4.4.1 Speech Detection

As soon as an audio clip is recorded, we segment the clip into smaller segments of equal length. Each segment, which is 3-second long, is the basic audio processing unit. Through experimentation we find this duration to be acceptable for the trade-off between inference delay and inference accuracy. It also captures adequately the turn-taking pattern normally present in everyday conversations [38]. This choice is also supported by previous studies showing that the median utterance duration of telephone conversations between customers of a major U.S. phone company and its customer service representatives is 2.74 seconds [60].

The result of the segmentation of an audio clip S is a sequence of N different segments, $S = \{S_1, S_2, ..., S_N\}$. Next we filter out segments containing long periods of silence or where noise is predominant. We use each segment's pitch value for this

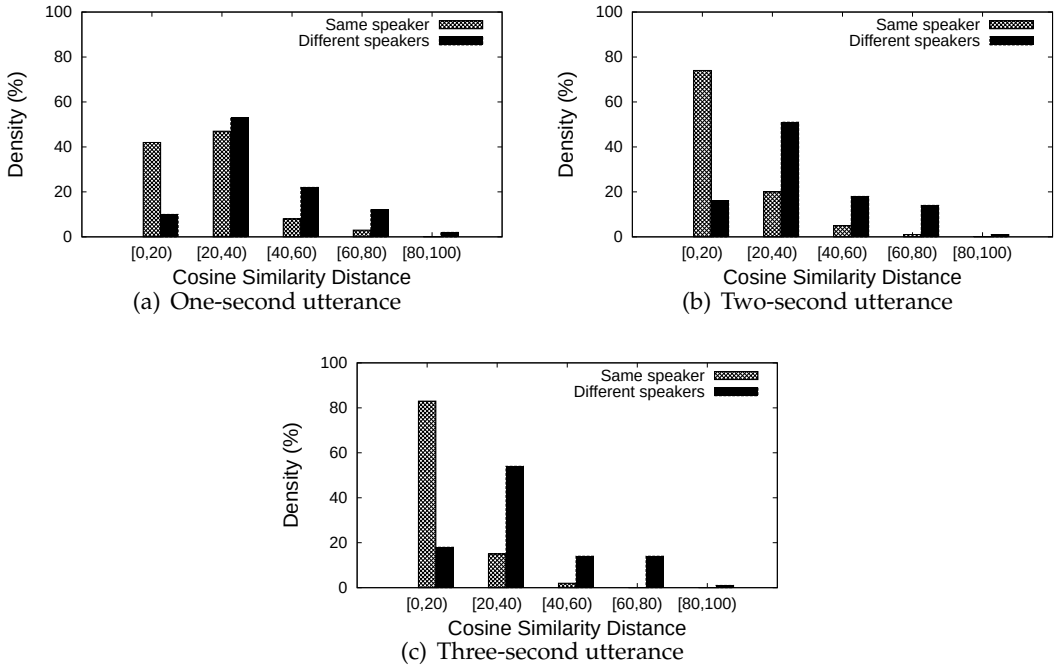


Figure 4.2: Cosine similarity distance demonstrates better speaker distinguishing capabilities with longer utterance.

purpose.

Pitch [64] is directly related to the speaker's vocal cord, and therefore, by being intimately connected with the speaker vocal trait, it's robust against noise and other external factors. Pitch has been widely used in speaker identification [7] and speaker trait identification [36] problems. When estimated accurately, pitch information can be used to assist the voice activity detection task in a noisy acoustic environment. In this study, we select YIN [12], a time-domain pitch calculation algorithm based on autocorrelation. While some other pitch estimation algorithms, such as Wu [75] and SAcC [33], might exhibit better accuracy, YIN is simpler, more energy-efficient, and robust to noise – hence more suitable for mobile devices.

Traditionally, energy-based methods such as the ones discussed in [20] have been used for voice data detection, but they are unsuitable for processing audio collected by smartphones. When recording audio, smartphones are usually placed at a certain distance from the speakers. As a result, even in absence of speech, the ambient audio energy could be high enough to trigger false positives in energy-based algorithms.

Pitch, on the other hand, is a better alternative because human pitch is distinctly different than pitch obtained in absence of speech.

We then apply the pitch estimation algorithm on all the segments to only admit those where the pitch falls within the range of 50 to 450 Hz, the typical pitch interval for human voice [6]. In this way, we apply a filtering technique to remove all the segments with long periods of silence or background noise. We note that using pitch to detect speech is not always the best approach because of pitch being only associated with voiced phoneme. However, in our setting, each basic acoustic segment is 3 seconds long with a probability of lack of voiced parts in such a time frame being quite low. In our evaluation, we collected over 1200 minutes audio and verified that pitch is a feasible solution for our purposes.

4.4.2 Speaker Distinguishing Features and System Calibration

Having filtered out the non-speech and background noise audio segments, our next step is to extract the features that can efficiently distinguish speakers. We have explored various feature sets that are largely used in the speech processing community, such as LPCC [40], RASTA [22], and different combinations of them. We find that MFCC [15] and pitch, when used together, provide the best inference results. In the following, we discuss the details on how these feature vectors are used in our counting algorithm.

MFCC and its Distance Metric

MFCC is one of the most effective and general-purpose features in speech processing [15]. In Crowd++, we use the coefficients between the 2nd and the 20th coefficient in order not to model the DC (direct current) component of the audio from the first coefficient. A 19-dimensional MFCC vector is then formed out of each 32 msec frame.

In order to perform the counting, we need to rely on a distance metric that allows Crowd++ to distinguish speech from different speakers by comparing MFCC vectors from different audio segments. An ideal distance metric should demonstrate a perfect

discriminative capability when computed on data from two different speakers. After investigating several common distance metric options – e.g., Average Linkage and 2-Gaussian Mixture Model (GMM) Generalized Likelihood Ratio (GLR)[1] – we find that Cosine Similarity (CS) is the best candidate as it minimizes the computation overhead in terms of real-time factor (RTF), defined as the processing time per second, and the expected error probability (EEP) metric. The EEP is defined as:

$$\int_{-\infty}^{\tau} p\left(x|\omega_d\right) dx + \int_{\tau}^{\infty} p\left(x|\omega_s\right) dx,$$

where $p\left(x|\omega_s\right)$ and $p\left(x|\omega_d\right)$ represent, respectively, the probability density functions (pdfs) of the distance from the same speaker and different speakers, and τ is the data point where these two pdfs present the same value. Table 4.1 shows that the best performance for both the RTF and EEP metrics is achieved using CS. This confirms the superiority of the CS distance compared to a GMM approach, heavily used in the literature in audio processing applications.

Distance Model	EEP	RTF
Cosine Similarity (CS)	0.1687	0.003
Average Linkage (AL)	0.5787	0.01
2-Gaussian Mixture Model (GMM)	0.5742	1.17

Table 4.1: Cosine Similarity outperforms Average linkage and 2-Gaussian Mixture Model in terms of expected error probability (EEP) and real time factor (RTF) based on 3-second utterances.

For the audio data processing, we partition the data into smaller segments, and assume the speech within a segment belongs to the same speaker. We then calculate the MFCC vectors for each segment and determine whether two segments belong to the same speaker by looking at their distance. We plot the cosine similarity distance density with different segment lengths (1, 2, 3 seconds) in Figure 4.2. We observe that the size of the overlap decreases as the length of the segment increases, which confirms the intuition that it is easier to distinguish multiple speakers when longer samples are collected. Finally, Figure 4.2 also provides hints about the best possible CS distance

[1]We use 2-GMM because a higher order GMM fails to converge in the parameter fitting phase.

threshold that allows the differentiation of different speakers.

Pitch and Gender Identification

In addition to assisting the speech detection process as discussed above, pitch can also be used to identify the gender of the speaker because the most distinctive trait between male and female voices is their fundamental frequency or pitch. The average pitch for men falls between 100 and 146Hz, whereas for women it is usually between 188 and 221Hz, as demonstrated in [6]. By relying on gender identification, Crowd++ speaker count accuracy is increased because of its disambiguation role. For instance, if two participants (one male and the other female) present similar MFCC features, their pitch difference can help distinguish between the two.

4.4.3 Crowd++ Counting Engine

The last step is about the computation of the speaker count. Having extracted n different audio segments containing human voice, Crowd++ derives the feature vectors from each segment. Let $M_1, M_2, ..., M_n$ be the sequence of feature vectors for all the segments, where M_i is the MFCC feature vectors for segment S_i.

Our counting algorithm involves two rounds. In the first round, we aggregate neighboring segments that produce similar features. Traditional speech processing methods use agglomerative hierarchical clustering [27] that requires the comparison between each segment with every other segment in the set, which incurs a computational complexity of $O(n^2)$. We instead employ a much more lightweight clustering method, i.e., forward clustering, which needs to visit all the segments only once. In forward clustering, we start from segment 1 (i.e., S_1), and compare it against S_2. If their MFCC features are close enough, i.e., $d_{CS}(M_1, M_2) < \theta_s$, we merge these two segments into a new S_1. Next we compare this new S_1 with S_3. If they are still similar, we will merge them too. Otherwise, we stop comparing with S_1, and begin to compare S_3 and S_4. In contrast with hierarchical clustering, forward clustering incurs much less computation and energy overhead given its linear time complexity $O(n)$. The

rationale behind forward clustering is that there usually exists temporal correlation in speech – the likelihood of contiguous segments containing the same voice is high when the segments are short enough. After running the forward clustering algorithm, we have fewer and longer segments, to the result of the merging step. We also note that longer segments have better performance in distinguishing different speakers and further boost counting accuracy.

Let's now denote with C the set of inferred speakers. When computing the distance $d_{CS}(i, j)$ between two different feature vectors M_i (which is the MFCC vector from a new segment i) and N_j (which is the MFCC vector of a previously inferred speaker, C_j) we have three possible outcomes:

- *Existing Speaker:* If $d_{CS}(i, j) < \theta_s$ and we infer a same gender, then we treat these two voice segments as belonging to the same person, namely C_j. In this case, we do not update C by adding new inferred speakers, but only update C_j's MFCC vector as M_i. If this condition is true for multiple existing speakers, we update the MFCC of the speaker that gives the lowest CS distance.

- *New Speaker:* If $d_{CS}(i, j) > \theta_d$ or different genders are inferred for all the members in C, we then tag this voice data as from a new speaker, the $|C| + 1$-th speaker, and add it to the admitted crowd C, where $|C|$ denotes the size of C.

- *Uncertainty:* If $d_{CS}(i, j) \geq \theta_s$ for all j's but $d_{CS}(i, k) \leq \theta_d$ for some k (both $j, k \leq |C|$), then we cannot decide whether this utterance is from an existing speaker or a new speaker. In this case, we discard this data point.

The θ_s and θ_d thresholds are empirically determined in the calibration phase before we conduct the evaluation. We note that the optimal threshold values may vary across different phone models because the microphones have different internal sensitivity levels. The choice of these two thresholds is driven by the desire to be conservative in the discovery of new speakers while minimizing the number of false positives.

To summarize, our counting algorithm is designed to be robust and resource-aware.

To this end, we rely on an energy efficient and noise-resilient pitch estimation algorithm, and introduce the cosine similarity distance function, an efficient distance metric at the core of our counting engine.

4.5 Evaluation

A detailed description of the Crowd++ evaluation results is presented in this section.

4.5.1 Crowd++ App Implementation

We have implemented Crowd++ on the Android platform using Java and installed it on multiple smartphones – HTC EVO 4G, Samsung Galaxy S2, S3, Google Nexus 4, – and tablets – Samsung Galaxy Tab 2 and Google Nexus 7. The raw audio is recorded at an 8 KHz frequency, 16 bit pulse-code modulation (PCM). We use 32 msec hamming window with 50% overlap for computing the MFCC, and the YIN pitch tracker. The code base of Crowd++ has been optimized to minimize the CPU processing time and energy consumption.

4.5.2 Energy Considerations

In Table 4.2, we report the latency for processing 1-second audio segments in terms of MFCC and pitch computation, and the time needed to run the speaker count algorithm on the different devices. The results show that Crowd++ execution time is fast, topping 320 msec and only 171 msec on a Galaxy S3. In addition, we demonstrate Crowd++ energy efficiency in a continuous sensing scenario. We adopt the duty-cycling approach of recording for 5-minute followed by the speaker count algorithm and sleeping for T minutes. We choose the Galaxy S2 phone and plot in Figure 4.3 the phone's battery duration as a function of the sleep time T between consecutive recordings (similar results can be found for other devices). We observe that even with short sleeping intervals, i.e., 15 minutes, the phone can last up to 23 hours. All the measurements are collected with the WiFi service running in background on the phone. These battery durations are all compatible with the use of a phone in a normal daily routine. It has to be noted

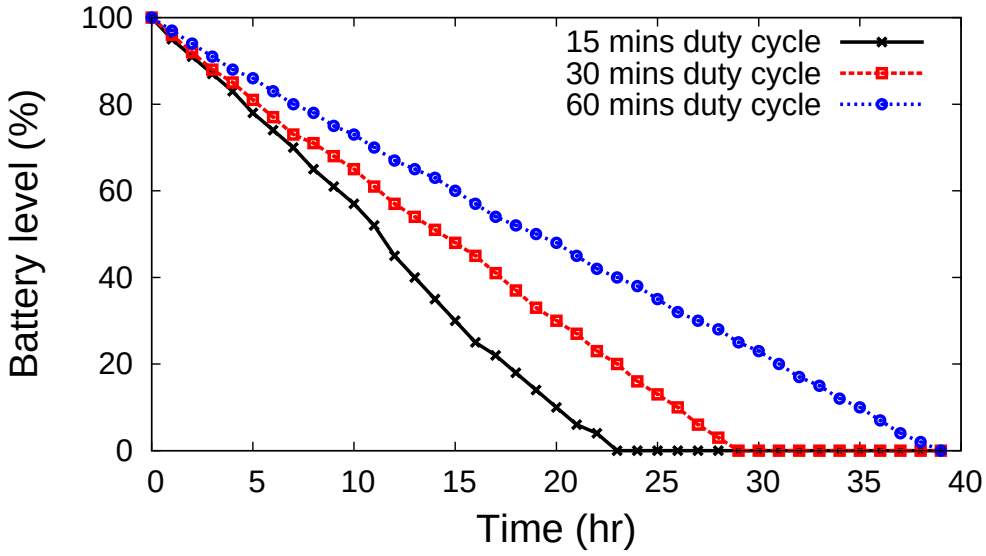


Figure 4.3: A duty-cycle of 15 mins guarantees a one day battery life for the Samsung Galaxy S2.

that these battery durations are achieved with a fixed duty-cycle policy, providing a performance lower bound. Given that Crowd++ would mostly run in public spaces only, longer sleeping intervals would extend the battery duration even further.

Latency (msec)	HTC EVO 4g	Samsung Galaxy S2	Samsung Galaxy S3	Google Nexus 4	Google Nexus 7
MFCC	42.90	36.71	24.41	22.86	23.14
Pitch	102.71	80.36	58.11	47.93	58.33
Count	175.16	150.47	89.01	83.53	70.23
Total	320.77	267.54	171.53	154.32	151.7

Table 4.2: Average latency for processing 1-second audio for MFCC calculation, Pitch calculation, and speaker counting using different phone models.

4.5.3 Performance Metric

We define *Error Count Distance* as $|\hat{C} - C|$, where C is the actual number of speakers and $\hat{C}$ is the estimated speaker count. The metric is calculated using the absolute value of the error to avoid the terms canceling out because of their positive and negative contributions. The average error count distance is a proxy for the Crowd++ count accuracy.

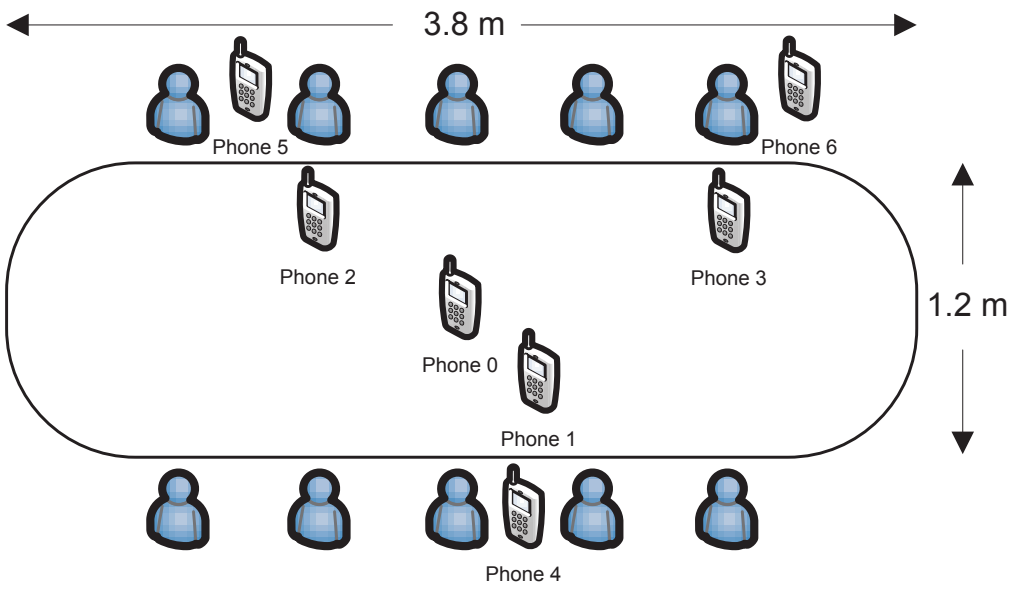


Figure 4.4: The phone placement in the benchmark experiments.

4.5.4 System Calibration

Before the feature computation of an arbitrary speech segment, we first need to set appropriate values for the parameters required by the CS metric to properly operate. For this purpose, we have performed a preliminary calibration phase at the beginning of the study, where we collect audio from 10 participants (5 males and 5 females) from different countries with different accents. To guarantee robust calibration, we use different phone models mentioned earlier with different placements (on the table and in the pocket), different distances (in a range of 2 meters), and orientations with respect to the speaker. We empirically chose 15 and 30, respectively, for the θ_s and θ_d thresholds used by the cosine similarity distance metric introduced in the previous section. θ_s and θ_d are chosen as the median value from $p\left(x|\omega_s\right)$ and $p\left(x|\omega_d\right)$, which is a little off from τ mentioned earlier to filter out the speech containing overlap and pause.

4.5.5 Performance with a Single Group of Speakers

We first conduct a set of controlled experiments to benchmark the performance of Crowd++. The experiment consists of 10 different sessions. The first session includes

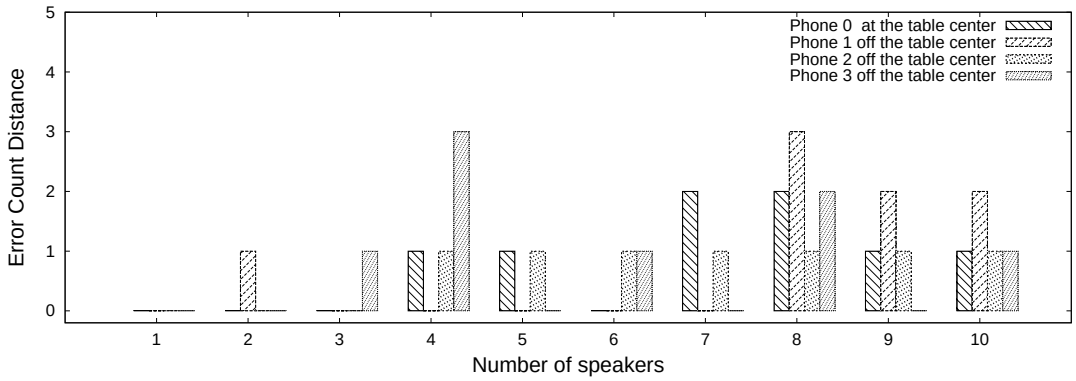


Figure 4.5: The counting accuracy does not vary much with the phone position on the table.

one speaker, and the number of speakers is incremented by 1 in each following session. As a result, the 10th session includes 10 speakers.

In each session, every speaker sits at an oval table and speaks in turns as in a conversation. Figure 4.4 illustrates the experimental setting. We use 7 smartphones for the audio recording – one smartphone (phone 0) is placed at the center of the table; 3 smartphones (phones 1-3) are placed on the table at a distance of 0.5 m, 1 m, and 1.5 m from the center; 3 smartphones (phones 4-6) are placed inside speakers' pockets.

Counting Accuracy vs. Phone Position

During a conversation, phones are usually placed on the table. Therefore, we look at how the phone's position on the table affects the error count. The results are shown in Figure 4.5. The results show that Crowd++ is rather robust against various conversation group sizes and phone positions. The error count distance is usually within 1, sometimes 2, and very rarely 3 (in 2 out of 40 cases). From this set of results, we can draw the following conclusions: First, in a quiet indoor environment, Crowd++ gives accurate speaker count estimates. Second, the phone's position on the table does not have an obvious impact on the inference accuracy.

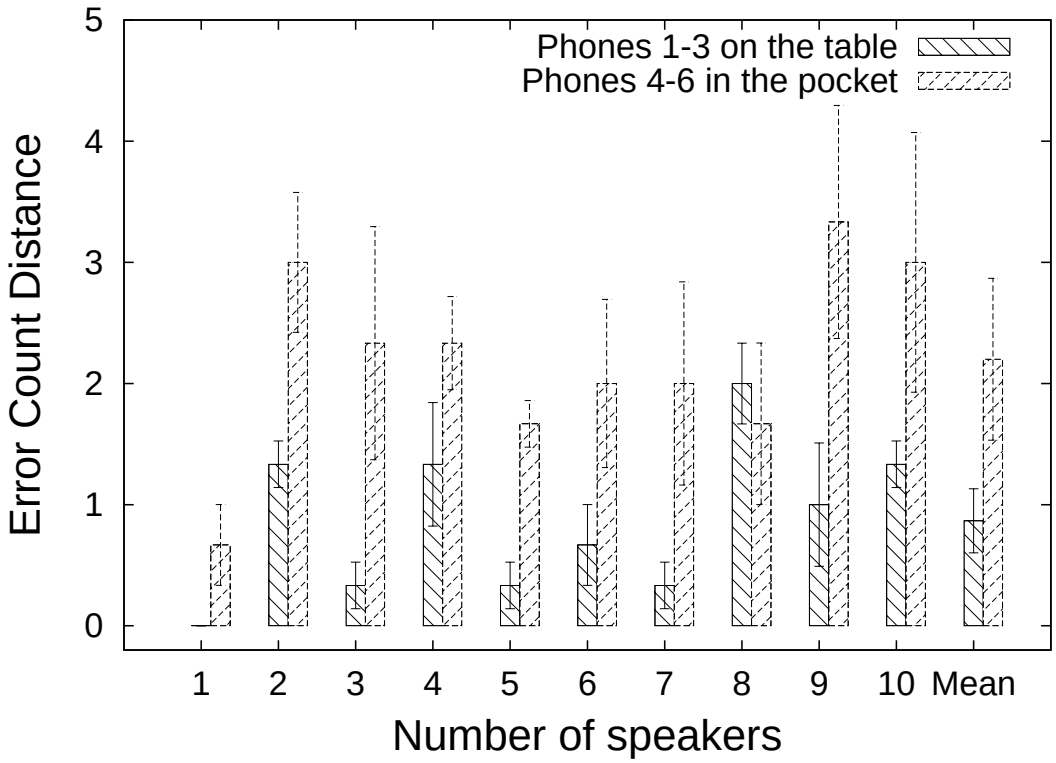


Figure 4.6: The phones on the table present a better counting accuracy than the phones inside the pockets.

Counting Accuracy when Phones on Table vs. in Pocket

Figure 4.6 compares the mean error count distance for phones placed on the table (namely, phones 1-3) and phones placed inside a pants pocket (namely, phones 4-6).

We find that in general, phones placed inside a pocket provide larger error count distances, similar to the trend observed in earlier smartphone-based audio sensing studies [42]. As a result, we suggest that in order to achieve accurate speaker count estimates, users should place their phones on the table to extend the sensing range of the microphone.

Counting Accuracy with Different Aggregation Methods

Given the proximity to the speakers, multiple phones record audio at the same time. We exploit this redundancy and compare different ways of aggregating the results. Specifically, we collect the speaker count estimates from all the 7 phones and show the

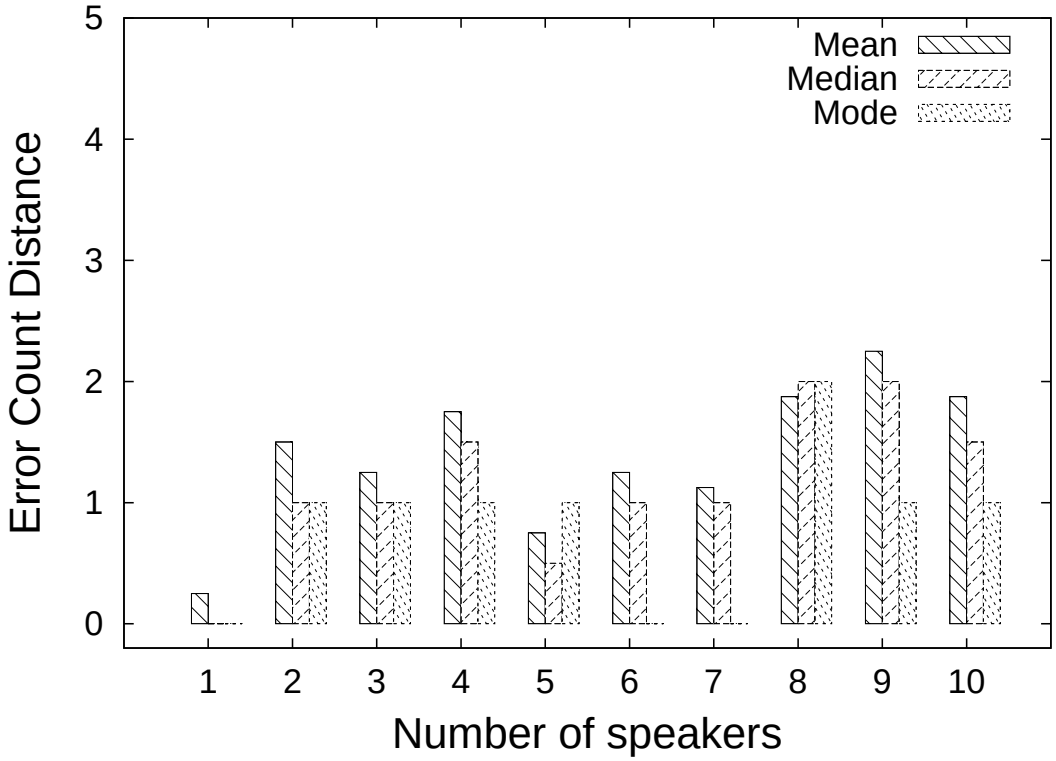


Figure 4.7: We achieve better count results when using median or mode from all the devices.

mean, median and mode of all the samples in Figure 4.7. The results show that the median and mode value give better speaker count estimates because they are more robust to estimation errors, and mode is better than the median in most of the cases.

4.5.6 Performance with Multiple Groups

We now investigate the performance of Crowd++ when operating in an environment where different groups of people are next to each other. This is the case of a restaurant for example, with each table occupied by a number of people. In this case, the speech from a nearby group could impact the results of the counting.

In order to demonstrate that Crowd++ can work in such a scenario, we have conducted another benchmark experiment to mimic a restaurant setting by having two and three groups of people talking at adjacent tables in the same room. Each group entertains separate conversations occurring in parallel. For each group, two phones are

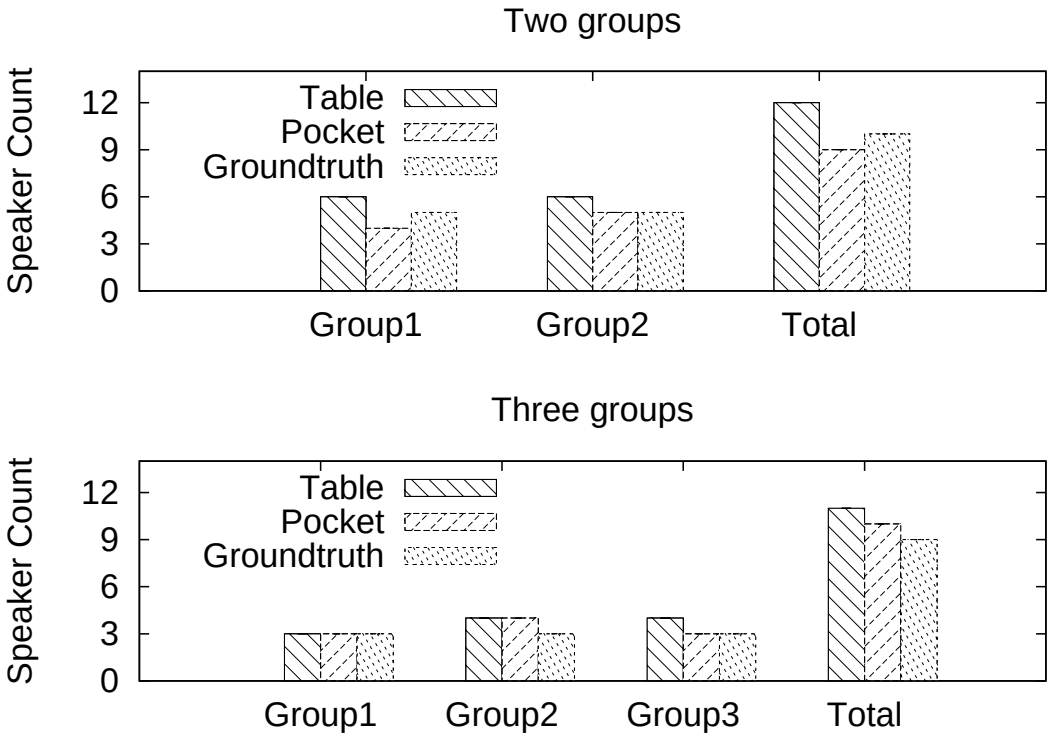


Figure 4.8: The phones inside the pockets present better counting results when multiple groups of speakers are co-located.

deployed: the first one held by one speaker and the second one in another speaker's pocket. In the two-group scenario, each group has 5 participants, while in the three-group scenario each group has 3 participants. The groups are separated by a 3-meter gap. We record 3 audio clips in each scenario.

We show the estimated speaker count in Figure 4.8. It is interesting to see that when we have multiple groups talking at the same time, the phones in the pocket have a slightly better performance. This is because the phone in the pocket is still able to pick up the group members' voice while filtering out – for the clothing muffling effect – more distant sounds.

Overall, both the phones in each group are able to accurately estimate the speaker count, with an average error distance of 1.5. It is important to realize that only 1 device is sufficient in a group of people to infer the speaker count.

In order to estimate the total number of people in a restaurant, our solution involves having each group estimate its size, and then using the sum as the total people count. In this experiment, we also evaluate the performance of this solution, which is shown in Figure 4.8 as well. We find that the average error count distance is reported 1 and 2 for the phones placed in the pockets or on the table. As a result, we believe that our divide-and-conquer solution works well in practice, especially considering the privacy concerns involved in uploading the audio features to the cloud for aggregation.

4.5.7 Performance with Various Conversation Parameters

In reality, many factors could impact the counting performance, such as utterance length, overlapping speech, and the duration of the recorded audio clip. Precisely controlling these parameters at the same time in real world experiments is often unfeasible. For this reason, we follow a common approach in the speech community and generate a separate dataset, as previously shown in [48]. Specifically, we collect audio recordings from 4 male and 4 female participants using a smartphone. We ask each speaker to talk for 3 minutes and record the audio clips. We then segment these clips into smaller segments of random lengths and assemble them to generate audio data.

We model the utterance length as a random variable following a log-normal distribution with mean of δ and standard deviation of 1 according to the configurations used in [60]. By default, each generated audio clip has 2, 4, 6, or 8 speakers, is 8 minutes long, no overlap, and is assigned a value of $\delta = 3$.

Counting Accuracy with Audio Clip Duration

In this set of experiments, we vary the audio duration from 2, to 4, 6 and 8 minutes. We report the average error count distances with these different audio durations in Figure 4.9. The results show that to achieve a good counting accuracy, we need longer audio clips. As shown in the plot, 8-minute audio clips are usually long enough to

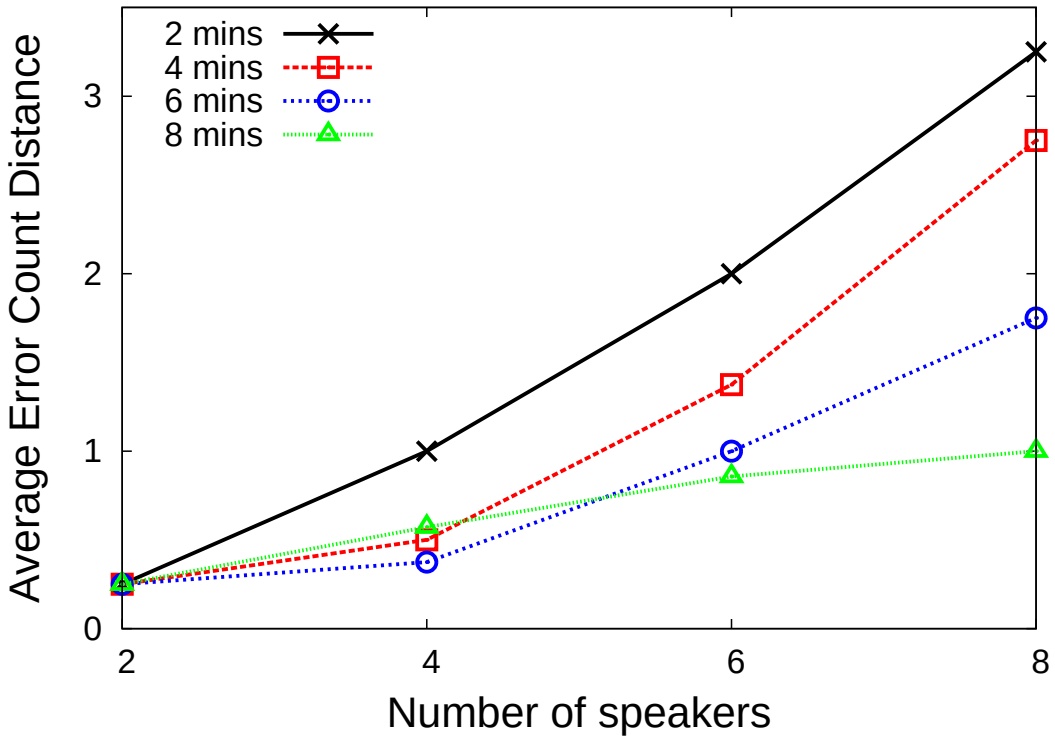


Figure 4.9: Eight-minute audioclips are sufficient to achieve an error count distance of 1.

achieve an average error count distance of 1. This is meaningful since we target the inference in social spaces, where usually people tend to remain for more than 8 minutes.

Counting Accuracy with Overlapping Percentage

Earlier studies [9] show that conversations are often characterized by interruptions of one speaker to another. In this set of experiments, we look at the impact of the percentage of the overlapping speech. We vary the overlapping percentage from 0%, 5%, 10%, 20% to 40% and show the results in Figure 4.10. We find that the overlapping percentage does not have a noticeable impact on the performance of Crowd++. Even with overlaps of 40% the average error distance of Crowd++ is about 1.

Counting Accuracy with Utterance Length

In daily conversations, utterance duration can vary according to the setting: people tend to be interrupted more frequently in casual chats and less in formal meetings. We

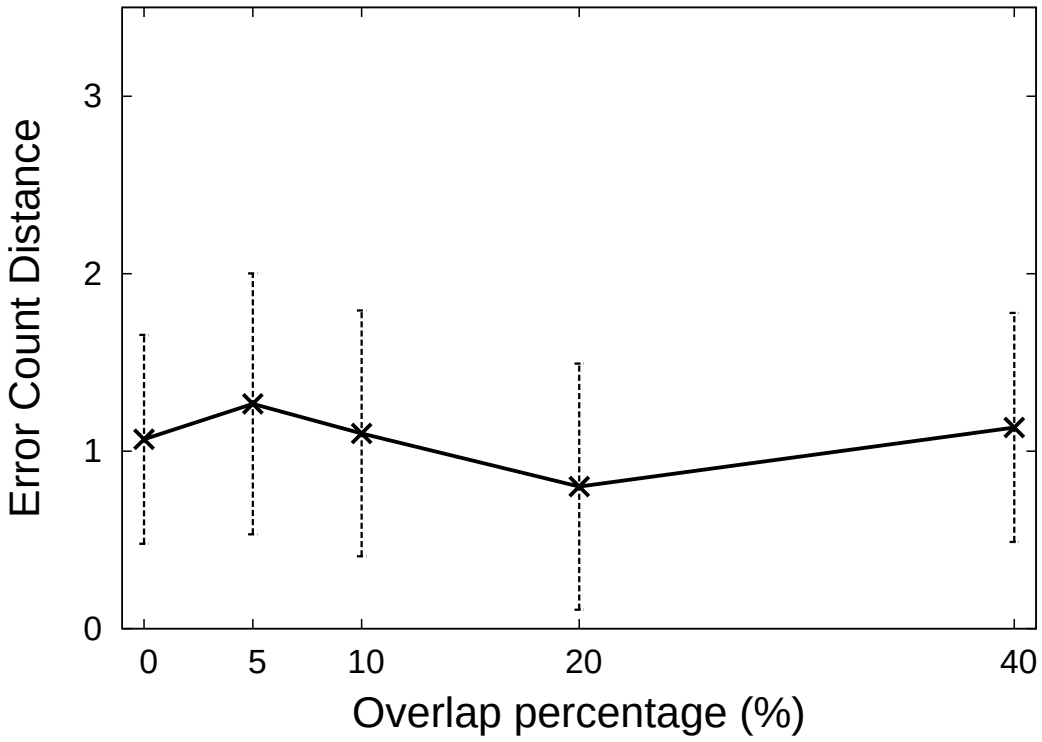


Figure 4.10: The average counting error distance is around 1 with up to 40% overlap.

then look at the impact of the utterance length, which we make it of 1, 2, 3, 5, and 8 seconds. The average error count distance is shown in Figure 4.11. We observe that we have slightly worse results when the utterance length is 1 or 2 seconds, shorter than the processing unit of 3 seconds. Even so, the average error distance is 1.5. When the utterance length is longer than 3 seconds, the average counting error distance decreases to 1.

4.5.8 Large-scale Experiments

To demonstrate that Crowd++ can accurately count speakers in different conditions, we have installed the app in six android devices and recruited six volunteers to collect 109 different audio instances with 120 people speakers for a total of 1034 minutes of recorded audio. The conversations are recorded during normal family and friend interactions. In each setting, the participants are within 1 meter from the phone. We can broadly group the audio clips into three categories based on the location of the

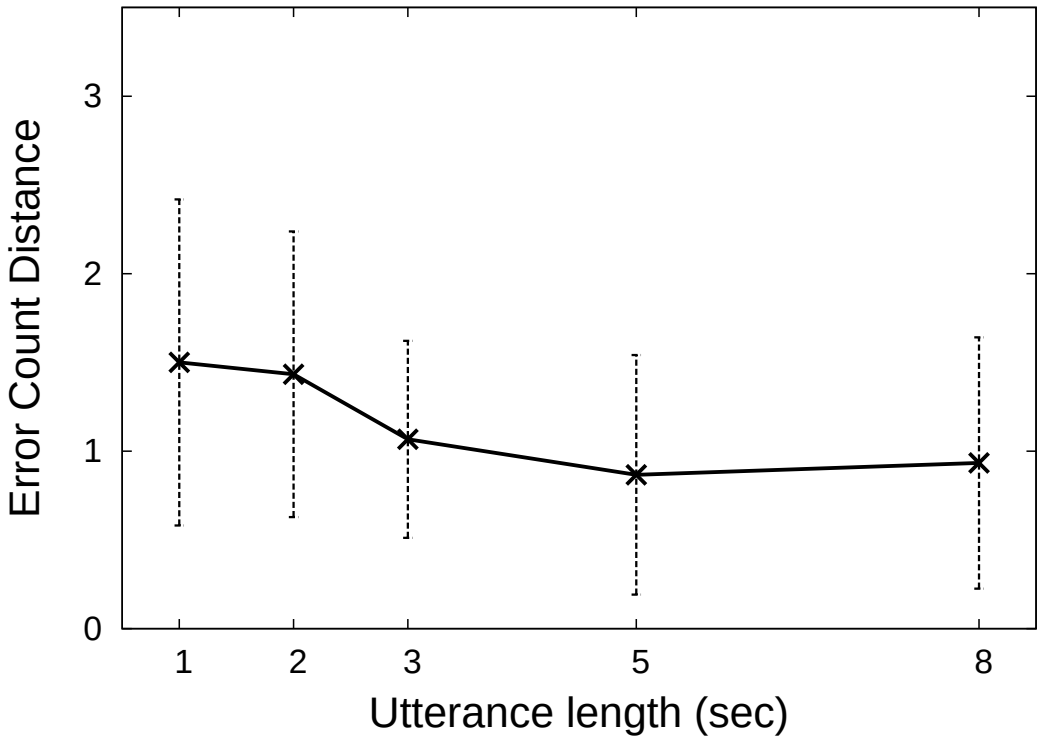


Figure 4.11: Longer utterance lengths lead to slightly better counting performance.

recordings:

- *Private Indoor Environments:* In this category, audio clips are recorded in quiet indoor environments, including seminar rooms, office and home during, respectively, different events: meetings, lunch and home conversations. The phones are placed on the table during the recording. In spite of these conversations taking place in private indoor settings, background noise is still present, for example paper flipping, door's closing/opening, chair movement, etc.

- *Public Indoor Environments:* In this scenario, audio clips are recorded in different public indoor environments when participants are sitting in restaurants, food courts or moving in supermarkets and shopping malls. The phones are placed in the pocket. The background noise in these environments is mainly generated by surrounding people, music, and various service operations.

- *Outdoor Environments:* The last class of recordings are collected in outdoor places

such as parking lots and restaurant outdoor seats, where the background noise mainly comes from cars, wind and other activities. The phones are placed in the pocket during recordings.

Private Indoor Environments					
Speaker #	Sample #	Place	SNR	AECD	AECP
2	4	Home	21	0	0%
3	11	Office	24.6	0.82	27.3%
4	8	Office	21.4	1.25	31.3%
5	7	Kitchen	20.9	1.28	25.6%
6	10	Kitchen	17.6	2	33%
Overall	40	Quiet Indoor	21.5	1.07	23.4%
Public Indoor Environments					
Speaker #	Sample #	Place	SNR	AECD	AECP
2	2	Restaurant	8.6	0	0%
3	6	Food Court	13.2	1.5	50%
4	7	Coffee Shop	8.2	1.86	46.5%
5	17	Shopping Mall	12.2	1.82	36.4%
7	12	Super Market	13.8	1.58	22.6%
Overall	44	Noisy Indoor	11.2	1.35	31.1%
Outdoor Environments					
Speaker #	Sample #	Place	SNR	AECD	AECP
2	4	Plaza	16.8	0.5	25%
3	6	Parking Lot	16.6	1.2	40%
4	7	Plaza	13	2.29	57.3%
5	2	Parking Lot	12.2	2.5	50%
6	6	Patio	13.9	2.67	44.5%
Overall	25	Noisy Outdoor	14.5	1.83	43.4%

Table 4.3: The detailed breakdown of the error counts for all the audio clips. We observe that average error count distances and average error count percentage for private indoor is less than in public indoor, and outdoor environments.

Table 4.3 summarizes the signal-to-noise ratio (SNR) estimation [28], average error count distance (AECD) and the average error count percentage (AECP)[2] from all the experiments. We observe a lower SNR and a higher AECD and AECP, when we move from private indoor, to public indoor and outdoor environments. We also observe that the error count increases when the crowd becomes larger because of more conflicting audio sources. The maximum AECD in private indoor scenarios is 2. In more challenging environments, e.g., public indoors and outdoors, the accuracy degrades.

[2]An alternative counting performance metric, defined as $\frac{|\hat{C}-C|}{C}$.

Being Crowd++ designed to infer social hotspots mainly indoors, we conjecture that the indoor error range can be considered adequate for many applications.

4.5.9 Crowd++ Use Cases

Finally, to demonstrate the utility of Crowd++, we have implemented three proof-of-concept use cases where knowing the number of speakers in a conversation is important.

Where Is the Most Crowded Restaurant?

Crowd++ can be exercised to find the most crowded restaurant in the area. To provide such a service, we envision there is at least one smartphone at each table running Crowd++, which counts the number of people talking at the table. Then we calculate the total number of people in the restaurant by summing up the people at each table.

In this study, we have recruited participants to record audio at four different restaurants, including formal restaurants, coffee shops, food courts in a mall and in a university student center. The results are shown in Table 4.3 in the public indoor environments section, where AECD is 1.3. We believe that this level of accuracy should be adequate for this use case since, again, our goal is to infer an estimate of the count in a lightweight manner. More accurate results could be achieved in cooperation with other techniques.

Are you a social person?

In the second use case, Crowd++ can be used to build a person's social diary – how many people the person talks to, and at what time – which is particularly useful for seniors or people with clinic depression.

In this study, we have recruited three participants, a teacher, a student, and a company employee, who have been using the SocialDiary app to record their conversation log for a week. The SocialDiary app records audio every 2-hour for 8 minutes. We

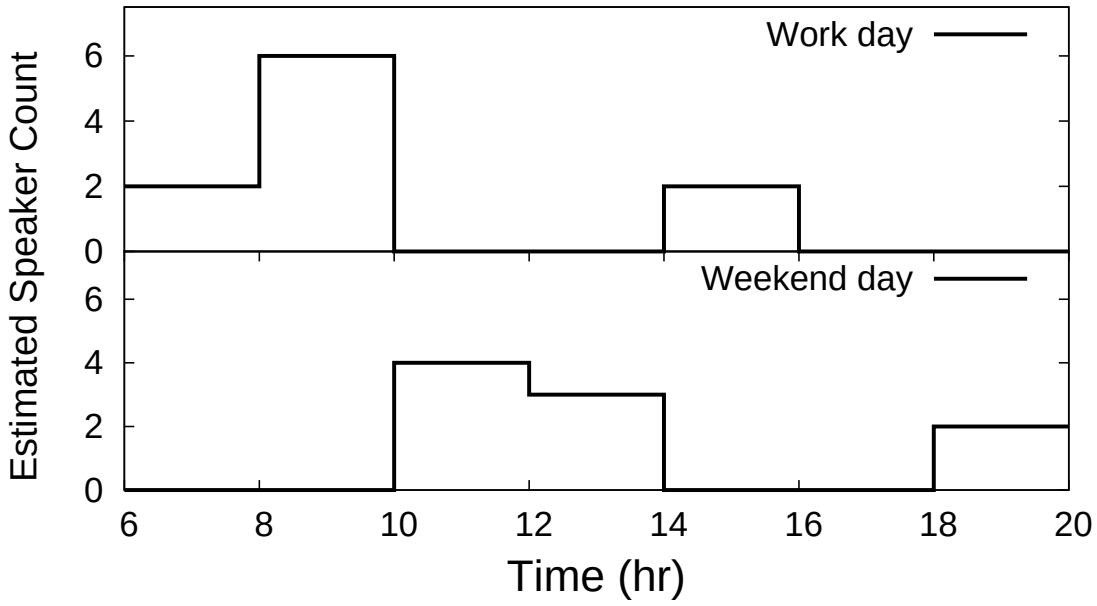


Figure 4.12: The social diary of a participant shows that he has different social patterns on work days and weekends.

show the log for the student participant on a weekday and a weekend in Figure 4.12. From the social diary, we observe that the student's day starts much later on the weekend. Also, he talks more in the morning and early afternoon on a weekday and at more recreational events – lunch and dinner – during weekends.

Is your audience engaged?

In the third use case, we show that Crowd++ can be used to measure the level of interaction of a lecture or seminar. Such a measurement could allow parents to be aware of their kids' participation in a classroom for example. Or it can be used to annotate a seminar or lecture to fast forward to the part with more active discussions when watching a video or audio recording of the event for example.

In this study, we record 2 regular classes and 2 recitation sessions at a university campus and 4 seminars from an industry lab. Each recording lasts 60 minutes. We segment each audio into six 10-minute segments and estimate the speaker count in each segment, as well as the total speaker count for the whole period. We show how the speaker count varies as time progresses in Figure 4.13. We observe that the recitation

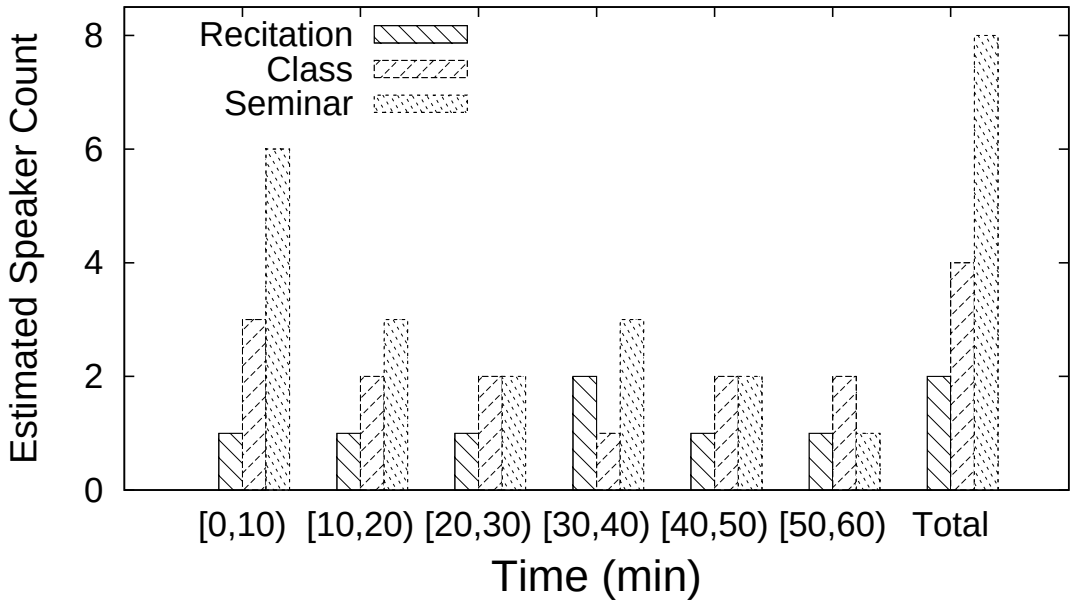


Figure 4.13: Seminars and classroom lectures have different interaction patterns over the time.

involves less interaction of all – the instructor spends most of the time showing how to solve the homework problems on the blackboard. The regular class has a steady interaction level throughout the duration, while the seminar presents more questions at the beginning.

4.6 Discussion

A possible source of interference is voice generated by TV or radio equipment. Would these background voices cause Crowd++ to over-count? The answer is no. Given the audio modulation techniques applied to TV and radio broadcast it has been proven that audio segments dominated by TV or radio sources can be effectively filtered out [37]. When there is instead significant overlap between people's live voices and TV or radio audio, source separation can be performed [35].

We acknowledge that in some cases the accuracy of Crowd++ could be improved; however, Crowd++ has been designed to perform people counting on mobile devices with no infrastructure intervention and in an energy and resource-efficient manner. Because of this, Crowd++ doesn't rely on complex speech processing algorithms that

would yield higher accuracy; its design favors efficiency and mobility support. We are currently exploring further optimizations, such as sparse sampling techniques to reduce the computation overhead. Moreover, noise cancellation achieved with the multiple microphones that can be found in most recent smarpthones would likely provide a further accuracy boost.

4.7 Related Work

We review the most relevant literature on smartphone audio inference applications and speaker count techniques.

4.7.1 Audio Sensing and Inference on Smartphones

A large body of research demonstrates the use of the mobile phone's microphone to opportunistically analyze audio for event and context characterization. Examples of smartphone context-aware applications are Darwin [42] and SpeakerSense [38] to perform speaker identification. SurroundSense [4] analyzes audio events for place fingerprinting. The EmotionSense project [55] demonstrates the possibility to classify humans' emotions through audio analysis. Ambient noise is leveraged to improve indoor localization results in [65]. All these projects have often in common the use of cumbersome supervised learning approaches, the use of external hardware in some cases, and the need to rely on external servers to operate the learning process. In contrast, Crowd++ is entirely unsupervised, with sensing and processing entirely occurring on the mobile device itself.

4.7.2 Speaker Counting

Some speaker counting techniques can be found in the literature. The closest related research to Crowd++ is [2] and [48]. Agneessens et al. [2] present a pitch estimation algorithm to recognize a single speaker from audio recordings containing two speakers with 70% of the times correctly estimate the speaker count (referred to as counting accuracy). Crowd++ goes beyond this binary classification approach by tackling a

much harder problem, where the number of speakers is up to 10 or even more. Moreover, Crowd++ runs an unsupervised learning algorithm without taking any training data from the target speakers. Ofoegbu et al. [48] present 60% counting accuracy for 4 speakers (versus Crowd++'s 68% counting accuracy under the same conditions and settings) and a generalized residual radio algorithm with a computational complexity of $O(N^2)$ (versus Crowd++'s $O(N)$). Moreover, the data set in [48] is based on staged data from the HTIMIT database [57] containing transcribed speech of American English speakers. Crowd++'s focus instead is the analysis of audio recordings challenged by noise, mobility and obstacles as people go about their daily lives. Another relevant technique is speaker diarization [3], which essentially determines "who spoke when" in an audio recording that contains an unknown amount of speech and also an unknown number of speakers. However, the main objective of diarization is to cluster the homogeneous speech rather than determine the optimal number of clusters. In addition, it usually relies on computationally expensive models (GMM, HMM) and algorithms (BIC, MCMC), which are not suitable for off-the-shelf smartphones.

4.8 Conclusion and Future Work

In this chapter, we presented Crowd++, a scalable and energy efficient speaker count application for smartphones based on the microphone's audio analysis. Crowd++ is novel in many dimensions: it is completely unsupervised and no prior models or external hardware are necessary to operate. It doesn't require any infrastructure and runs entirely on the mobile device. We implemented Crowd++ on different Android platforms and showed, through solid experimentation, that Crowd++ presents adequate inference accuracy in many diverse conditions, from quiet to noisy environments. In contrast to more complex and less scalable counting techniques, Crowd++ is a lightweight approach that can support many different application scenarios: from social sensing – to determine social hotspots – to personal wellbeing assessment and social diary, place characterization, and more accurate localization techniques.

Chapter 5

Conclusion

5.1 Summary

In conclusion, this dissertation investigates unobtrusive human context learning techniques. Specifically, we make the following contributions:

- *PC-DfP:* By observing and utilizing how human interferes with radio frequency based wireless signal in indoor environments, we designed and developed a wireless embedded networked sensor system to track an individual's location, trajectory and speed without requiring him/her to carry any devices. We developed approaches that mitigate the errors caused by the multipaths in indoor radio propagation and reduce the human calibration effort.

- *SCPL:* Next, based upon how multiple people disturb the radio signals when they co-exist in an indoor environment, we designed a set of algorithms to count the number of people and localize them using the calibration data collected with only one person. In addition, we incorporated the map information, explored the human mobility constraints in typical indoor settings to improve the tracking accuracies.

- *Crowd++:* We designed and implemented a mobile application that records a conversation, extracts the speaker-independent features and automatically counts the number of speakers in an unsupervised manner. This new context information (speaker count) enables a number of social sensing applications such as crowd estimation, social isolation detection and event engagement level estimation.

5.2 End Note

Looking forward, with the vision of "internet of things", there will be numerous small form-of-factor devices anywhere in our ambient living environments. They are not only coming with their own utilities to improve our daily life, but also leading to new sensing modalities. On the other side, the increasing number of such ambient devices should facilitate easier human intervention. Thus, the benefit of a technology would not come at the cost of inconvenience to the end users. While this dissertation takes the first few steps in this direction, enriching applications in this domain remains an open problem. Sensing gesture, gait, fall and emotion in unobtrusive manners are of great value, and the ability to accurately detect them is one important step towards realizing technology-assisted life style and wellbeing. For example, gesture detection is important as people send commands to and interact with others through gestures; the study of gait allows diagnoses and intervention strategies to be made to people who have problems walking, as well as permitting future developments in technology-enhanced rehabilitation; a fall can lead to severe injuries, and even death, especially for elders, and thus it is extremely important to develop device-free techniques for fall detection; a person's emotional state has a great impact on his/her wellbeing, and the ability to detect what causes a person's emotional swing using mobile computing will be important to explore. Ultimately, all the immersive ambient technology will transparently monitor our behavior, adapt to our needs, and help actively manage our life.

References

[1] Owl platform: The great owl watches all things. http://www.owlplatform.com/.

[2] A. Agneessens, I. Bisio, F. Lavagetto, M. Marchese, and A. Sciarrone. Speaker count application for smartphone platforms. In *IEEE ISWPC*, 2010.

[3] X. Anguera Miro, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals. Speaker diarization: A review of recent research. *IEEE Transaction on Audio, Speech and Language Processing*, 20(2), 2012.

[4] M. Azizyan, I. Constandache, and R. Roy Choudhury. Surroundsense: mobile phone localization via ambience fingerprinting. In *ACM MobiCom*, 2009.

[5] P. Bahl and V. Padmanabhan. Radar: an in-building rf-based user location and tracking system. In *IEEE INFOCOM*, 2000.

[6] R. Baken. *Clinical measurement of speech and voice*. College-Hill Press, 1986.

[7] M. J. Carey, E. S. Parris, H. Lloyd-Thomas, and S. Bennett. Robust prosodic features for speaker identification. In *ICSLP*, 1996.

[8] D. Cassioli, M. Win, and A. Molisch. A statistical model for the uwb indoor channel. In *IEEE VTC*, 2001.

[9] O. Cetin and E. Schriberg. Speaker overlaps and asr errors in meetings: Effects before, during, and after the overlap. In *IEEE ICASSP*, 2006.

[10] A. B. Chan, Z.-S. Liang, and N. Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *IEEE CVPR*, 2008.

[11] X. Chen, A. Edelstein, Y. Li, M. Coates, M. Rabbat, and A. Men. Sequential monte carlo for simultaneous passive device-free tracking and sensor localization using received signal strength measurements. In *ACM/IEEE IPSN*, 2011.

[12] A. D. Cheveigné and H. Kawahara. Yin, a fundamental frequency estimator for speech and music. *The Journal of the Acoustical Society of America*, 111(4), 2002.

[13] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan. Indoor localization without the pain. In *ACM MobiCom*, 2010.

[14] T. Choudhury and A. Pentland. Sensing and modeling human networks using the sociometer. In *IEEE ISWC*, 2003.

[15] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 28(4), 1980.

[16] D. De, W.-Z. Song, M. Xu, C.-L. Wang, D. Cook, and X. Huo. Findinghumo: Real-time tracking of motion trajectories from anonymous binary sensing in smart environments. In *IEEE ICDCS*, 2012.

[17] E. Elnahrawy, X. Li, and R. P. Martin. Using area-based presentations and metrics for localization systems in wireless lans. In *IEEE ICN*, 2004.

[18] B. Firner, C. Xu, R. Howard, and Y. Zhang. Multiple receiver strategies for minimizing packet loss in dense sensor networks. In *ACM MobiHoc*, 2010.

[19] G. D. Forney Jr. The viterbi algorithm. *Proceedings of the IEEE*, 61(3), 1973.

[20] J. Haigh and J. Mason. Robust voice activity detection using cepstral features. In *TENCON*, 1993.

[21] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, 2nd edition, 2003.

[22] H. Hermansky and N. Morgan. Rasta processing of speech. *IEEE Transactions on Speech and Audio Processing*, 2(4), 1994.

[23] T. W. Hnat, E. Griffiths, R. Dawson, and K. Whitehouse. Doorjamb: unobtrusive room-level tracking of people in homes using doorway sensors. In *ACM SenSys*, 2012.

[24] D. B. Jayagopi, H. Hung, C. Yeo, and D. Gatica-Perez. Modeling dominance in group conversations using nonverbal activity cues. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(3), 2009.

[25] O. Kaltiokallio, M. Bocca, and N. Patwari. Enhancing the accuracy of radio tomographic imaging using channel diversity. In *IEEE MASS*, 2012.

[26] P. G. Kannan, S. P. Venkatagiri, M. C. Chan, A. L. Ananda, and L.-S. Peh. Low cost crowd counting using audio tones. In *ACM SenSys*, 2012.

[27] G. Karypis, E.-H. Han, and V. Kumar. Chameleon: Hierarchical clustering using dynamic modeling. *IEEE Computer*, 32(8), 1999.

[28] C. Kim and R. M. Stern. Robust signal-to-noise ratio estimation based on waveform amplitude distribution analysis. In *INTERSPEECH*, 2008.

[29] K. Kleisouris, B. Firner, R. Howard, Y. Zhang, and R. P. Martin. Detecting intra-room mobility with signal strength descriptors. In *ACM MobiHoc*, 2010.

[30] P. Krishnan, A. Krishnakumar, W.-H. Ju, C. Mallows, and S. Gamt. A system for lease: location estimation assisted by stationary emitters for indoor rf wireless networks. In *IEEE INFOCOM*, 2004.

[31] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer. Multi-camera multi-person tracking for easyliving. In *IEEE VS*, 2000.

[32] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.

[33] B. S. Lee and D. P. W. Ellis. Noise robust pitch tracking by subband autocorrelation classification. In *INTERSPEECH*, 2012.

[34] B. Liu, Y. Jiang, F. Sha, and R. Govindan. Cloud-enabled privacy-preserving collaborative learning for mobile sensing. In *ACM SenSys*, 2012.

[35] G. Liu, D. Dimitriadis, and E. Bocchieri. Robust speech enhancement techniques for asr in non-stationary noise and dynamic environments. In *INTERSPEECH*, 2013.

[36] G. Liu, Y. Lei, and J. H. Hansen. A novel feature extraction strategy for multi-stream robust emotion identification. In *INTERSPEECH*, 2010.

[37] G. Liu, C. Zhang, and J. H. Hansen. A linguistic data acquisition front-end for language recognition evaluation. In *Odyssey*, 2012.

[38] H. Lu, A. B. Brush, B. Priyantha, A. K. Karlson, and J. Liu. Speakersense: energy efficient unobtrusive speaker identification on mobile phones. In *Pervasive*, 2011.

[39] D. Madigan, E. Einahrawy, R. Martin, W.-H. Ju, P. Krishnan, and A. S. Krishnakumar. Bayesian indoor positioning systems. In *IEEE INFOCOM*, 2005.

[40] J. E. Markel and A. H. Gray. *Linear Prediction of Speech*. Springer-Verlag, 1982.

[41] A. Matic, V. Osmani, and O. Mayora. Automatic sensing of speech activity and correlation with mood changes. *Pervasive and Mobile Sensing and Computing for Healthcare*, 2012.

[42] E. Miluzzo, C. T. Cornelius, A. Ramaswamy, T. Choudhury, Z. Liu, and A. T. Campbell. Darwin phones: the evolution of sensing and inference on mobile phones. In *ACM MobiSys*, 2010.

[43] R. S. Moore, R. Howard, P. Kuksa, and R. P. Martin. A geometric approach to device-free motion localization using signal strength. Technical report, Technical Report, Rutgers University, 2010.

[44] M. Nakatsuka, H. Iwatani, and J. Katto. A study on passive crowd density estimation using wireless sensors. In *ICMU*, 2008.

[45] S. Nannuru, Y. Li, M. Coates, and B. Yang. Multi-target device-free tracking using radio frequency tomography. In *IEEE ISSNIP*, 2011.

[46] S. Nannuru, Y. Li, Y. Zeng, M. Coates, and B. Yang. Radio frequency tomography for passive indoor multi-target tracking. *IEEE Transactions on Mobile Computing*, PP(99), 2012.

[47] L. Ni, Y. Liu, Y. C. Lau, and A. Patil. Landmarc: indoor location sensing using active rfid. In *IEEE PerCom*, 2003.

[48] U. O. Ofoegbu, A. N. Iyer, R. E. Yantorno, and B. Y. Smolenski. A speaker count system for telephone conversations. In *IEEE ISPACS*, 2006.

[49] R. J. Orr and G. D. Abowd. The smart floor: a mechanism for natural user identification and tracking. In *ACM CHI*, 2000.

[50] N. Patwari and J. Wilson. Rf sensor networks for device-free localization: Measurements, models, and algorithms. *Proceedings of the IEEE*, 98(11), 2010.

[51] J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland, and T. Carey. *Human-computer interaction*. Addison-Wesley Longman Ltd., 1994.

[52] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *ACM MobiCom*, 2000.

[53] M. Rabbi, S. Ali, T. Choudhury, and E. Berke. Passive and in-situ assessment of mental and physical well-being using mobile sensors. In *ACM UbiComp*, 2012.

[54] L. Rabiner and B.-H. Juang. *Fundamentals of speech recognition*. Prentice-Hall, Inc., 1993.

[55] K. K. Rachuri, M. Musolesi, C. Mascolo, P. J. Rentfrow, C. Longworth, and A. Aucinas. Emotionsense: a mobile phones based adaptive platform for experimental social psychology research. In *ACM UbiComp*, 2010.

[56] T. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall PTR New Jersey, 2nd edition, 2001.

[57] D. A. Reynolds. Htimit and llhdb: Speech corpora for the study of handset transducer effects. In *IEEE ICASSP*, 1997.

[58] B. Ristic, S. Arulampalm, and N. Gordon. *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House Publishers, 2004.

[59] T. Roos, P. Myllymaki, and H. Tirri. A statistical modeling approach to location estimation. *IEEE Transactions on Mobile Computing*, 1(1), 2002.

[60] A. E. Rosenberg, A. Gorin, Z. Liu, and S. Parthasarathy. Unsupervised speaker segmentation of telephone conversations. In *INTERSPEECH*, 2002.

[61] B. Schilit, N. Adams, and R. Want. Context-aware computing applications. In *IEEE Workshop on Mobile Computing Systems and Applications*, 1994.

[62] M. Seifeldin and M. Youssef. A deterministic large-scale device-free passive localization system for wireless environments. In *ACM PETRA*, 2010.

[63] A. Smailagic and D. Kogan. Location sensing and privacy in a context-aware computing environment. *IEEE Wireless Communications*, 9(5), 2002.

[64] K. Sonmez, E. Shriberg, L. Heck, and M. Weintraub. Modeling dynamic prosodic variation for speaker verification. In *ICSLP*, 1998.

[65] S. P. Tarzia, P. A. Dinda, R. P. Dick, and G. Memik. Indoor localization without infrastructure using the acoustic background spectrum. In *ACM MobiSys*, 2011.

[66] F. Thouin, S. Nannuru, and M. Coates. Multi-target tracking for measurement models with additive contributions. In *IEEE FUSION*, 2011.

[67] M. Valtonen, J. Maentausta, and J. Vanhala. Tiletrack: Capacitive human tracking using floor tiles. In *IEEE PerCom*, 2009.

[68] R. Want, A. Hopper, V. Falcão, and J. Gibbons. The active badge location system. *ACM Transactions on Information Systems*, 10(1), 1992.

[69] M. Weiser. The computer for the 21st century. *Scientific american*, 265(3), 1991.

[70] J. Weppner and P. Lukowicz. Collaborative crowd density estimation with mobile phones. In *ACM PhoneSense*, 2011.

[71] J. Wilson and N. Patwari. Radio tomographic imaging with wireless networks. *IEEE Transactions on Mobile Computing*, 9(5), 2010.

[72] J. Wilson and N. Patwari. See-through walls: Motion tracking using variance-based radio tomography networks. *IEEE Transactions on Mobile Computing*, 10(5), 2011.

[73] J. Wilson and N. Patwari. A fade level skew-laplace signal strength model for device-free localization with wireless networks. *IEEE Transactions on Mobile Computing*, 11(6), 2012.

[74] K. Woyach, D. Puccinelli, and M. Haenggi. Sensorless sensing in wireless networks: Implementation and measurements. In *IEEE WiOpt*, 2006.

[75] M. Wu, D. Wang, and G. J. Brown. A multipitch tracking algorithm for noisy speech. *IEEE Transactions on Speech and Audio Processing*, 11(3), 2003.

[76] C. Xu, B. Firner, R. S. Moore, Y. Zhang, W. Trappe, R. Howard, F. Zhang, and N. An. Scpl: indoor device-free multi-subject counting and localization using radio signal strength. In *ACM/IEEE IPSN*, 2013.

[77] C. Xu, B. Firner, Y. Zhang, R. Howard, and J. Li. Trajectory-based indoor device-free passive tracking. In *IWMS*, 2012.

[78] C. Xu, B. Firner, Y. Zhang, R. Howard, J. Li, and X. Lin. Improving rf-based device-free passive localization in cluttered indoor environments through probabilistic classification methods. In *ACM/IEEE IPSN*, 2012.

[79] C. Xu, M. Gao, B. Firner, Y. Zhang, R. Howard, and J. Li. Towards robust device-free passive localization through automatic camera-assisted recalibration. In *ACM SenSys*, 2012.

[80] C. Xu, S. Li, G. Liu, Y. Zhang, E. Miluzzo, Y.-F. Chen, J. Li, and B. Firner. Crowd++: Unsupervised speaker count with smartphones. In *ACM UbiComp*, 2013.

[81] M. Youssef and A. Agrawala. Small-scale compensation for wlan location determination systems. In *IEEE WCNC*, 2003.

[82] M. Youssef and A. Agrawala. The horus wlan location determination system. In *ACM MobiSys*, 2005.

[83] M. Youssef, M. Mah, and A. Agrawala. Challenges: device-free passive localization for wireless environments. In *ACM MobiCom*, 2007.

[84] D. Zhang, Y. Liu, and L. Ni. Rass: A real-time, accurate and scalable system for tracking transceiver-free objects. In *IEEE PerCom*, 2011.

[85] D. Zhang, J. Ma, Q. Chen, and L. M. Ni. An rf-based system for tracking transceiver-free objects. In *IEEE PerCom*, 2007.

[86] D. Zhang and L. M. Ni. Dynamic clustering for tracking multiple transceiver-free objects. In *IEEE PerCom*, 2009.

Appendix A

Acknowledgment of Previous Publications

Chapter 2 revises previous publications [78, 79]:
C. Xu, B. Firner, Y. Zhang, R. Howard, J. Li, and X. Lin. Improving rf-based device-free passive localization in cluttered indoor environments through probabilistic classification methods. In ACM/IEEE IPSN, 2012.

Chapter 3 revises a previous publication [76]:
C. Xu, B. Firner, R. S. Moore, Y. Zhang, W. Trappe, R. Howard, F. Zhang, and N. An. Scpl: indoor device-free multi-subject counting and localization using radio signal strength. In ACM/IEEE IPSN, 2013.

Chapter 4 revises a previous publication [80]:
C. Xu, S. Li, G. Liu, Y. Zhang, E. Miluzzo, Y.-F. Chen, J. Li, and B. Firner. Crowd++: Unsupervised speaker count with smartphones. In ACM UbiComp, 2013.

Appendix B

Refereed Publications as a Ph.D. Candidate

Vijay Srinivasan, Saeed Moghaddam, Abhishek Mukherji, Kiran K. Rachuri, **Chenren Xu**, Emmanuel M. Tapia. MobileMiner: Mining Your Frequent Behavior Patterns on Your Phone. In ACM UbiComp, 2014.

Feixiong Zhang, Yanyong Zhang, Alex Reznik, Hang Liu, Chen Qian and **Chenren Xu**. A Transport Protocol for Content-Centric Networking with Explicit Congestion Control. In IEEE ICCCN, 2014.

Chenren Xu, Vijay Srinivasan, Jun Yang, Yoshiya Hirase, Emmanuel M. Tapia, and Yanyong Zhang. Context-aware Global Power Management for Mobile Devices Balancing Battery Outage and User Experience. In ACM HotMobile, Poster Session, 2014.

Vijay Srinivasan, Saeed Moghaddam, Abhishek Mukherji, Kiran K. Rachuri, **Chenren Xu**, and Emmanuel M. Tapia. On-device Mining of Mobile Users' Context Co-occurrence Patterns. In ACM HotMobile, Poster Session, 2014.

Chenren Xu, Sugang Li, Gang Liu, Yanyong Zhang, Emiliano Miluzzo, Yih-Farn Chen, Jun Li, and Bernhard Firner. Crowd++: Unsupervised Speaker Count with Smartphones. In ACM UbiComp, 2013.

Chenren Xu. Device-free People Counting and Localization. In ACM UbiComp Ph.D. Forum.

Chenren Xu, Bernhard Firner, Robert Moore, Yanyong Zhang, Wade Trappe, Richard Howard, Feixiong Zhang, and Ning An. SCPL: Indoor Device-free Multi-subject Counting and Localization Using Radio Signal Strength. In ACM/IEEE IPSN, 2013.

Feixiong Zhang, Alex Reznik, Hang Liu, **Chenren Xu**, Yanyong Zhang, and Ivan Seskar. Using ORBIT for Evaluating Wireless Content-centric Network Transport. In ACM WiNTECH Workshop in conjunction with ACM MobiCom, Demo Session, 2013.

Robert S. Moore, Bernhard Firner, **Chenren Xu**, Richard Howard, Yanyong Zhang, and Richard Martin. Building a Practical Sensing System. In IEEE iThings, 2013.

Robert S. Moore, Bernhard Firner, **Chenren Xu**, Richard Howard, Richard Martin, and Yanyong Zhang. It's Tea Time: Do You Know Where Your Mug Is? In ACM HotPlanet Workshop in conjunction with ACM SIGCOMM, 2013.

Ashwin Ashok, **Chenren Xu**, Tam Vu, Marco Gruteser, Richard Howard, Yanyong Zhang, Narayan Mandayam, Wenjia Yuan, and Kristin Dana. Bifocus: Using Radio-optical Beacons for an Augmented Reality Search Application. In ACM MobiSys, Demo Session, 2013.

Chenren Xu, Mingchen Gao, Bernhard Firner, Yanyong Zhang, Richard Howard, and Jun Li. Towards Robust Device-free Passive Localization Through Automatic Camera-assisted Recalibration. In ACM SenSys, Poster Session, 2012.

Chenren Xu, Bernhard Firner, Yanyong Zhang, Richard Howard, Jun Li, and Xiaodong Lin. Improving RF-based Device-free Passive Localization in Cluttered Indoor Environments Through Probabilistic Classification Methods. In ACM/IEEE IPSN, 2012.

Chenren Xu, Bernhard Firner, Yanyong Zhang, Richard Howard, and Jun Li. Exploiting Human Mobility Trajectory Information in Indoor Device-free Passive Tracking.

In ACM/IEEE IPSN, Poster Session, 2012.

Chenren Xu, Bernhard Firner, Yanyong Zhang, Richard Howard, and Jun Li. Trajectory-based Indoor Device-free Passive Tracking. In ACM/IEEE IPSN Workshop on Mobile Sensing, 2012.

Chenren Xu, Bernhard Firner, Yanyong Zhang, Richard Howard, and Jun Li. Statistical Learning Strategies for RF-based Indoor Device-free Passive Localization. In ACM SenSys, Poster Session, 2011.

Bernhard Firner, **Chenren Xu**, Richard Howard, and Yanyong Zhang. Multiple Receiver Strategies for Minimizing Packet Loss in Dense Sensor Networks. In ACM MobiHoc, 2010.