Spring 4-2016

Error Mitigation in RSSI-Based Fingerprinting Localization Using Multiple Communication Channels

Francisco Javier Mora-Becerra
University of Nebraska-Lincoln, pancho_2009@hotmail.com

ERROR MITIGATION IN RSSI-BASED FINGERPRINTING LOCALIZATION USING MULTIPLE COMMUNICATION CHANNELS

by

Francisco Javier Mora-Becerra

A THESIS

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfilment of Requirements

For the Degree of Master of Science

Major: Electrical Engineering

Under the Supervision of Professor Lance C. Pérez

Lincoln, Nebraska

April, 2016

ERROR MITIGATION IN RSSI-BASED FINGERPRINTING LOCALIZATION USING MULTIPLE COMMUNICATION CHANNELS

Francisco Javier Mora-Becerra, M.S.

University of Nebraska, 2016

Adviser: Lance C. Pérez

Location data from radio signal strength indication (RSSI) based wireless networks has been used in various applications such as creating smart home behavioral monitoring systems, tracking health care workers for the spread of hospital-associated infections, and providing location-aware tour guide systems. Because RSSI-based systems are inexpensive and can be used with most wireless devices without requiring additional hardware, they are a popular choice for localization. Unfortunately, multipath fading dramatically degrades the performance of an RSSI-based system's ability to locate a target indoors. This thesis endeavors to reduce localization error for RSSI-based fingerprinting localization systems in an indoor environment through frequency diversity by using multiple communication channels. By creating a multichannel fingerprint of the environment using fingerprinting calibration techniques, fine-grained, 5 centimeter, 2-dimensional localization accuracy is achieved in an indoor environment under certain restrictions.

DEDICATION

To my father. Gracias por todos los días que me obligaste a ayudarte con las reparaciones de la casa y trabajos de mecánica. Aunque a veces no me gustaba y odiaba hacerlo, cada día era una oportunidad nueva para enfrentar un reto y poder formar la mente de un ingeniero.

Contents

List of Figures

List of Tables

Chapter 1

Introduction to Localization Methods

Since the dawn of civilization, humans have been localizing – localizing things with hand-drawn maps, localizing each other with smoke signals, and localizing themselves across the sea, guided by the stars in the sky. Localization is the process of finding the position or location of a specific target based on some observable phenomena. People use localization for a glut of applications: localizing soldiers in combat, collecting marketing data, tracking endangered turtles, navigating self-driving cars, and even tracking battery packs on the international space station.

The most popular technology used for outdoor localization is the global positioning system (GPS). GPS is a space-based navigation system first implemented in 1973 that uses 31 satellites in orbit to provide location and time information in all weather conditions across the globe. A GPS receiver listens for time-stamped signals transmitted from the GPS satellites to compute propagation delay, then solves a set of equations using the computed distances to determine its physical location on earth. This is known as time-of-arrival (TOA) based localization.

GPS is freely available to anyone with a GPS receiver to use anywhere on or near the Earth. Receivers are now commonly found in cars, airplanes, ships, and smartphones. In addition to its initial military purpose, GPS-enabled devices have been used in many applications including physical activity tracking [49], transportation and logistics [57, 99], and rehabilitating patients with GPS-enabled wearable sensors [89]. Although GPS is a reliable outdoor localization technology, it suffers from dramatic performance degradation indoors because the microwave radio signals used by GPS are greatly attenuated by walls and ceilings [101]. Indoor GPS technology exists, but this technology is extremely expensive due to significant processing requirements [28]. For these reasons, indoor localization remains an active research field and a reliable low-cost indoor localization solution still eludes the research community.

1.1 Indoor Localization Methods

The five most common indoor localization methods are acoustic, inertial/mechanical, laser, computer vision, and radio frequency (RF) [74, 109]. RF methods include timing-based, angle-of-arrival, and received signal strength indication (RSSI).

1.1.1 Active Versus Passive Systems

Active and passive systems must first be defined before delving into further discussion on localization systems. A localization systems typically includes a *target*, i.e., the object to be localized, *anchors* which are transceivers placed in the environment with known fixed locations, a localization algorithm which makes locations estimate based on data from the target and anchors, and a performance metric to measure the system's prediction error. The terms passive and active refer to system characteristics that are defined with respect to the target. In active systems, the target device is actively transmitting a signal that allows the system (composed of anchors) to determine the target's location. In passive systems, the target listens for signals and determines its own location.

1.1.2 Acoustic Methods

Acoustic-based localization systems typically use electrical devices that either transmit or receive sonic waves, mechanical vibrations transmitted over a solid, liquid, or gaseous medium [109]. Systems can indirectly determine the distance between communicating devices by computing the distance traveled by the sonic wave by recording the time of transmission and taking into account the speed of sound. The most popular acoustic methods are ultrasonic-based localization systems which use waves typically above 20 kHz. Examples of ultrasonic systems include the Massachusetts Institute of Technology's (MIT) Cricket system and Cambridge University's Bat system [44]. In a passive configuration, MIT's Cricket system employs anchors distributed throughout a building that periodically transmit ultrasonic pulses. These pulses are received by a mobile target that then computes its own location using trilateration. Cambridge's Bat system is an active system where users wear small badges emitting ultrasonic pulses. The network of anchors then computes the 3D position of the badges through multilateration. These ultrasonic-based localization active systems are typically able to achieve sub-meter accuracy in an indoor environment under certain conditions [106].

An ultrasonic-based localization system must have direct line-of-sight between the anchors and mobile targets to avoid erroneous distance estimates due to computing distances for non line-of-sight paths. Furniture can cause such obstructions in an indoor environment. Sometimes anchors are placed on the ceiling of a room to help eliminate obstructions from furniture or other objects. Also, depending on the system, the anchor and mobile target orientation is important. In the Cricket system, ultrasonic transducers on the mobile targets must point in the general vicinity of the anchor transducers. This becomes an issue when tracking a human because the person may not always hold or wear the mobile transmitter/receiver in a position that meets these ideal conditions. Another contributing factor to localization error is the variation in the speed of sound for sonic wave propagation in air due to environmental changes such as temperature, humidity, and atmospheric pressure [48, 58], because the systems are dependent on the speed of sound to calculate the distance. Because of this, sonic wave-based systems cannot localize well in environments with frequent and drastic changes in temperature, humidity, and atmospheric pressure [109] unless the system includes these factors in its prediction model and uses additional sensors to measure them.

1.1.3 Inertial/Mechanical Methods

Inertial/Mechanical technologies can measure the mechanical movement energy that is exerted on to them [109]. Systems can measure the energy of the direct application of force on such technologies. For example, Orr et al. [85] have used metallic plates with load cells in a project called "Smart Floor" where the plates were laid on the ground and used to identify a person walking over them. Orr et al. performed tracking by recording every instance that a person walked over the plates at different locations. In addition to localization, researchers have used pressure sensing floor plates for fall detection of the elderly [6].

Another approach to measuring mechanical movement energy is via inertial sensors, typically accelerometers and gyroscopes. Thanks to advancements in microelectromechanical systems technology, small surface-mount sensor packages are commonly found in phones, smartwatches, and other mobile devices. But, because inertial sensors only yield relative positioning information and they produce noisy measurements due to inherent drift and measurement quantization, they are usually part of a hybrid localization system. Hybrid systems combine different technologies so that an additional source of position information serves as an absolute reference. Algorithms like the Kalman filters and particle filters [58] then use data fusion to make location estimates by integrating the information from these various sources. For example, activity data captured by accelerometer sensors has refined localization data from an RF-based system in [34, 80].

The primary issue with inertial sensors is the presence of a bias offset added to the measured signal causing a drift in the sensor's relative position information. Even in the absence of any input (including gravitational pull), inertial sensors output a non-zero value. This offset is dependent on time, temperature, and stochastic factors that occurs due to inherent mechanical properties of the sensor. These factors cannot be eliminated due to current limitations in manufacturing processes [39].

1.1.4 Photonic Methods

Photonic methods capture electromagnetic waves at a frequency within or near the human visible spectrum. Photonic energy refers to the energy carried by the electromagnetic radiation within visible light or the nearby ultraviolet and infrared (IR) spectra [109]. Several methods capture photonic energy and use it for localization, including laser range finders and cameras.

Laser range finders emit concentrated beams of light and sense the reflection that comes off of a wall or object to infer distance. There exist various techniques to infer distance from beam reflection measurements including phase-shift conversion [92], where laser systems modulate the emitted beam with either a square or sinusoidal waveform to be compared with the reflected wave which will have some small phase shift due to the time delay of the light beam propagation. The systems then associate the phase shift with a certain distance by considering the speed of light. High-end laser range finders like the Hokuyo UTM-30LX use this technique to yield up to one centimeter of accuracy indoors [47]. Similar laser systems have been used for simultaneous localization and mapping (SLAM) in robotic navigation [88, 104]. The systems provide fine grain localization and appear to be the best fit for active indoor localization, but the hardware involved is bulky, extremely expensive, and requires considerable data processing.

A popular passive photonic indoor localization method uses computer vision through mobile or fixed camera systems. High quality digital cameras have become ubiquitous thanks to advancements in camera technology and smart phones. Researchers have used mobile camera systems for SLAM assisted robotic navigation [3], user localization with the use of a cell phone camera [98], and self localizing smart backpacks for indoor environments [72]. There are typically two stages for localization with a mobile camera: an offline stage where the system collects visual features of the environment, such as structural features of the building or fiducial markers, and an online stage where algorithms use these features as a reference to compute location. Fixed camera systems take a different approach. They are usually mounted in the environment looking over an area. These camera systems can use feature extraction techniques (such as facial recognition) to provide localization and tracking solutions for security monitoring and surveillance [91].

The primary issues associated with camera based systems are that computer vision algorithms require high processing demands, a large storage capacity is needed to store images or video, and cameras are often considered an invasion of privacy when used for human tracking or localization.

1.1.5 Radio Frequency Methods

Indoor radio frequency (RF) localization methods estimate the location of a mobile target by measuring one or more properties of RF signals [109]. These methods typically rely on either timing measurements, angle-of-arrival (AOA), or radio signal strength indication (RSSI) [105]. Unlike GPS that uses long range satellites, indoor RF systems typically use short range local anchors that can be deployed indoors.

1.1.5.1 Radio Frequency Timing Methods

RF timing methods use measurements of the propagation delay of RF waves traveling through a medium between two communicating devices. In air, RF waves travel at the speed of light, i.e., three hundred million meters per second. Because of this, timing methods require expensive and complicated hardware for high timing resolution down to 0.5 nanoseconds to measure the travel time of an RF wave for half a foot of resolution [46]. Localization methods using RF timing use several such measurements to compute 2-D or 3-D positions with techniques like trilateration. In practice, these methods have inherent difficulties because precise clock synchronization across multiple devices is a major issue [112]. RF timing based systems are mainly distinguishable by their constraints on clock synchronization.

Three popular timing based systems are time-of-arrival (TOA), time-difference-of-arrival (TDOA), and roundtrip time-of-flight (RTOF) [68]. In active TOA based systems, TOA is a time measurement of the one-way propagation delay between the mobile target and the anchors. This requires precise time synchronization between the mobile target and all of the anchors, below 1 nanosecond for indoor localization accuracy in the decimeter range. Active TDOA based systems use the TDOA of received signals for localization. Here, TDOA is the difference in the times at which the signal arrives at multiple anchors, unlike the absolute arrival time of TOA [119]. The benefit of this is that only the anchors in the TDOA based systems require synchronization amongst each other. Systems can replace the absolute synchronization constraint with a less precise constraint than that of a RTOF based Systems [112]. Here, the mobile target transmits a signal then waits for the anchors to transmit it back to complete the roundtrip propagation delay measurement. The synchronization challenge for RTOF based systems is that the mobile target must know the exact delay needed for the anchors to resend the packet. Even a delay offset of 1 millisecond can correspond to measurement deviations of several meters for some systems.

1.1.5.2 Radio Frequency Angle-of-Arrival

Angle-of-arrival refers to the angle between the received signal of an incident wave and some reference direction [62]. The most common approach to identify the angular direction of the signal is through antenna diversity. Typically, antenna arrays on the receiving devices are used to determine the AOA for AOA-based localization systems. Once the AOA is measured, AOA-based localization systems use various localization methods like triangulation to identify the location of the target by solving a system of direct equations for intersecting lines [71].

1.1.5.3 Radio Signal Strength Indication

Radio signal strength indication is the distance dependent measurement of a received signal's power. RSSI presents itself at the front end of a receiver to determine amplification levels needed for demodulation. Typically RSSI is measured in dBm, which is ten times the base ten logarithm of the ratio between the power at the receiving end and the reference power [87]. Most radios oftentimes provide RSSI because it is directly related to the performance of communication schemes: low RSSI corresponds to poor wireless communication due to high bit-error-rate during the demodulation process.

The availability of RSSI measurements on most off-the-shelf radios helps stimulate the interest in designing RSSI-based ranging techniques [13, 73]. In an active system, local devices deployed in a room or building measure RSSI. Popular wireless network platforms used in RSSI-based systems include WiFi [96, 110], Bluetooth [12, 97, 107], and ZigBee [93].

TOA and AOA based systems typically achieve higher localization accuracy than RSSI-based systems. However, the amount of achievable accuracy also correlates with the hardware complexity and device cost [90]. AOA systems require multiple antennas that increases the size of the device [90]. TOA-based systems require high speed signal processing and have high device costs with high energy consumption [77, 119]. In contrast, RSSI-based ranging techniques are low cost because oftentimes they do not require additional hardware and they possess small computational requirements that do not burden the on board circuitry [73]. Additionally, RSSI-based localization systems are especially desirable because they are already available on most off-the-shelf commercial radios. For these reasons, people use RSSI-based systems for many applications including navigation assisted tours, behavioral monitoring, studying the spread of hospital related infections, tracking basketball players, navigation for underground mining, and localizing people and equipment for construction job-sites [22, 33, 50, 74, 93, 110].

Table 1.1 displays a summary of advantages and disadvantages for the aforementioned methods.

Technology	Advantages	Disadvantages
Ultrasonic	Sub-meter localization.	External synchronization. Speed of sound variations are dependent on temperature and other environmental conditions.
Inertial/Mechanical	Self-contained. Resilient to environmental conditions.	Inherent sensor drift. Relative localization. Requirement of initiation and calibration.
Laser	Location accuracy of about 3 cm.	Extremely expensive. High processing requirements
Computer Vision	High localization and orientation accuracy.	High processing requirements. Dependent on illumination conditions and environmental noise. Sensitive to obstructions and reflections.
RF Timing Methods	Sub-meter localization.	Expensive hardware required for precise synchronization. High processing requirements
RF Angle-of-Arrival	Meter localization.	High processing requirements.
Radio Signal Strength Indication	Usage of readily deployed equipment; reduced cost.	Coarse localization. Sensitive to interference, signal propagation effects, and dynamic environmental change.

Table 1.1: Technologies used for indoor localization [109]

1.2 Factors Affecting RSSI-Based Systems

All radio frequency waves undergo attenuation when they propagate through a medium. Propagation in air results in path-loss, or reduction in power density, for electromagnetic waves that is proportional to the distance traveled. In an active RSSI-based localization system, this decrease also limits the transmission range of the mobile target. RSSI-based systems rely on the distance dependent attenuation nature of RSSI to provide range information. The distance dependent line-of-sight path-loss model in air, or free space, is given by

$$\mathbf{RSSI}(d) = \mathbf{RSSI}_0 - 10n_p \log \frac{d}{d_0},$$

where d is the distance between the devices, $\mathbf{RSSI}_0$ is the RSSI measured at the reference distance d_0, and n_p is the path-loss exponent. Using this path-loss model, one can compute the distance between a transmitter and receiver if the RSSI is known. An active localization system that collects three RSSI measurements from three anchors could theoretically compute the 3-dimensional location coordinates for the mobile target using trilateration. Unfortunately there are various natural phenomena that alter the path-loss model including RF wave reflection and scattering. These phenomena result in multiple copies of the signal being received at each anchor, otherwise widely known as multipath propagation [4]. Multipath propagation causes rapid variations in the RSSI when communicating devices move short distances relative to each other due to constructive/destructive signal interference [43]. Multipath propagation is often seen in indoor environments where moving a small distance drastically changes RSSI. Figure 1.2.1 illustrates how moving an anchor from one location, labeled with the number one, to another location, labeled with the number two, changes the signal strength due to summing waves with different phases. The varying phases occur from signals traveling through multiple paths of different lengths. For the sake of simplification, the illustration shows two paths, one blue and one green, neither of which are non line-of-sight. In some cases where line-of-sight conditions cannot be met, line-of-sight systems will fail at localizing the target. These cases are commonly encountered indoors.

Other factors affecting RSSI and contributing to localization error include antenna orientation and ambient temperature. If the radiation patterns for the antennas used

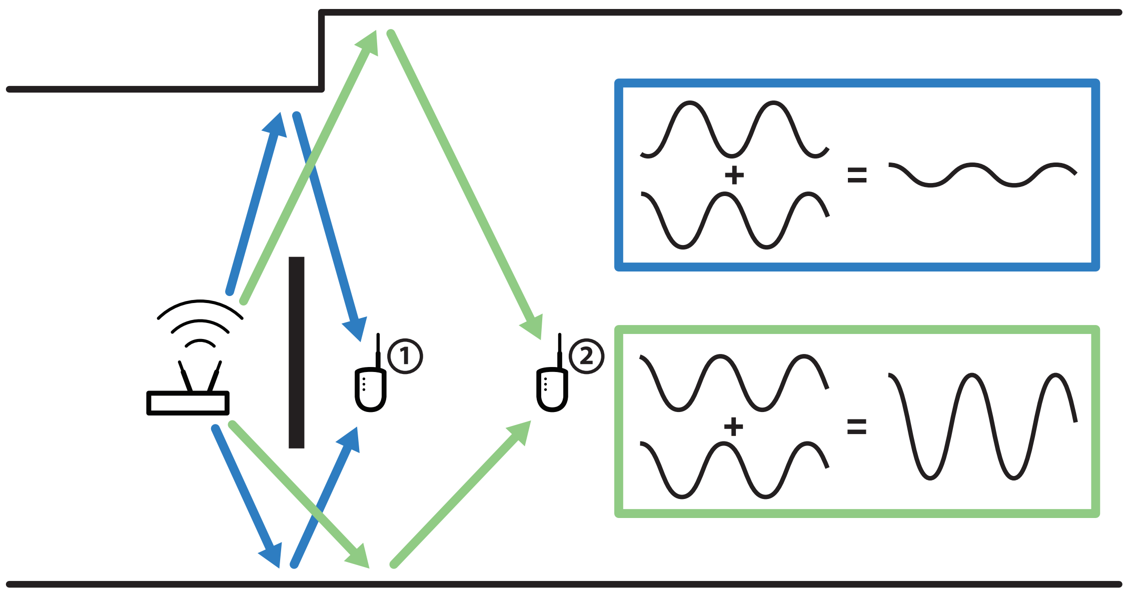


Figure 1.2.1: Multipath propagation effect on RSSI

in the system are not omnidirectional, which is often the case, then the orientation of the antenna will affect RSSI measurements [81]. Additionally, it has been observed that ambient temperature influences hardware performance in WSNs, resulting in altered RSSI measurements [17]. In particular, temperature changes can cause a shift of crystal frequency, increased thermal noise of the transceiver, and saturated amplifiers.

1.2.1 Multichannel RSSI-Based Localization for Multipath Effect Mitigation

One approach to improve localization accuracy for RSSI-based systems is to simultaneously measure RSSI data for different frequencies. The RSSI measured at a single location is affected by destructive or constructive interference from the superposition of RF waves from multipath propagation. When waves travel through multiple paths and meet at a single point, they will sum with varying attenuations and phases. The phases are frequency dependent, so varying the frequency will vary the observed RSSI for that single point. In complex indoor environments, this phenomenon is virtually unpredictable and too complicated to model. Thus, researchers often modify the path-loss model of RSSI in multipath environments as

$$\mathbf{RSSI}(d) = \mathbf{RSSI}_0 - 10n_p \log \frac{d}{d_0} + X_\sigma,$$

where X_σ is a random variable representing the erratic behavior of RSSI due to multipath propagation [13]. The random variable X_σ is assumed to have a Gaussian distribution with zero mean. In an attempt to eliminate the effect of this random variable X_σ, various measurements can be recorded on multiple communication channels and averaged. In this model, averaging mitigates the effect of multipath propagation on the RSSI. Various groups have shown that multichannel frequency averaging improves RSSI-based localization results [5, 13, 18, 66, 93]. Frequency averaging is just one of many frequency diversity methods.

1.2.2 Fingerprinting in RSSI-Based Localization

Fingerprinting is a technique of machine leaning that evolved from the study of pattern recognition and computational learning theory in artificial intelligence [37]. Machine learning explores the study and construction of algorithms that can learn from and make predictions on data [35]. In RSSI-based localization, fingerprinting algorithms infer location information of RF devices based on previously collected RSSI measurements. There are several reasons why people use fingerprinting algorithms in RSSI-based localization systems [11, 54, 86]. First, they can provide a solution to localization problems where traditional methods fail to deal with multipath propagation. Second, it is relatively easy to obtain an RSSI dataset that can be used by the fingerprinting algorithms. Third, low complexity fingerprinting algorithms like k-nearest neighbor perform well in practice [18].

Fingerprinting can prove advantageous when used with multichannel data. These techniques are able to treat input RSSI data separately, rather than simply averaging values. This is the motivation behind using fingerprinting is this work. RSSI fingerprinting is well documented in the subsequent chapters.

1.3 Problem Statement

RSSI-based localization systems are good candidates for indoor environments because they are low cost and do not require additional hardware. The disadvantage of using RSSI-based systems are the difficulties associated with unpredictable position-dependent RSSI measurements caused by multipath propagation. Traditional approaches [5, 13, 18, 66, 93] attempt to mitigate multipath propagation effects to improve localization accuracy through frequency averaging where RSSI is recorded over multiple communication channels and averaged for each position to approximate the RSSI of an environment without multipath propagation.

Unlike traditional methods that attempt to mitigate multipath propagation effects, here multipath propagation is used to advantage by creating a multichannel fingerprint of the environment. By sampling at a high enough spatial resolution, the system captures variations in the position-dependent RSSI. Since RSSI is frequency dependent in the indoor environment, the fingerprints will also be different for each frequency. It is hypothesized that capturing fingerprints of multiple frequencies will provide more information regarding the mobile target's location, which can be exploited to mitigate localization errors that affect single-frequency fingerprinting methods through frequency diversity.

This work uses three distinct algorithms to show that 2-dimensional localization accuracy is improved when using multiple frequencies for fingerprinting. The three methods are the k-nearest neighbor algorithm due to its low complexity, the state-of-the-art neural network as it is a method of choice in modern research [35], and the particle filter because it introduces a temporal component and allows for a problem definition using state-space and sampling of hypothesized error distributions. This work tests the algorithms on real RSSI data collected from a custom wireless sensor network using a single target and anchor. The results demonstrate that performance increases for all three algorithms as the number of frequencies is increased. The comparison determines which algorithm works best for indoor localization.

Chapter 2

Review of RSSI-Based Localization Methods

People have been researching and developing RSSI-based localization methods for two decades. The ability to measure RSSI with off-the-shelf radios – and its associated low cost – makes the technique desirable. This chapter serves as an introduction to various RSSI-based localization methods; it presents current work in the field while illustrating the most prominent challenges for this type of system. First, multipath propagation, the most prominent challenge in RSSI-based localization, is explained. Second, terminology for RSSI-based localization, is introduced. This includes a discussion on popular RSSI-based localization methods. Finally, the chapter concludes with a discussion of frequency diversity for error mitigation in RSSI-based localization.

2.1 The Multipath Propagation Model

Multipath propagation is a natural phenomenon of RF wave propagation that occurs when a transmitted RF signal reflects from objects in an environment and arrives at a destination via multiple paths. The reflections can originate from furniture, walls, people and other objects in an environment. From a receiver's point of view, the received signal is the superposition of all the signals traveling via the multiple paths [95]. Each signal varies in amplitude and phase depending on the distance traveled and number of reflections. The superposition of the multiple signals may result in constructive or destructive interference.

Multipath propagation is especially apparent indoors and difficult to model due to the presence of objects and furniture in the room. Additionally, varying room shape and size creates various unpredictable signal propagation paths. Other factors such as absorption coefficients and scattering effects add more complexity to the model. Bardella et al. [13] state that an extremely accurate channel model would require perfect knowledge of the environment, and further mentions that such a model would lack generality and reusability. For this reason, it is not practical to create a complete model of multipath propagation experienced in a RSSI-based localization systems to mitigate the localization error.

To better understand multipath propagation, consider the case where a receiver (RX) and transmitter (TX) are placed in the same environment at the same height as shown in Figure 2.1.1. In an active system, the mobile *target* is the transmitter and the *anchor* is the receiver. The mobile target is continuously sending a signal with a fixed frequency and amplitude, while the anchor receives the signal and makes an RSSI measurement. The anchor is set to a fixed location but the mobile target moves freely to or from the anchor while maintaining the same height. Now assume

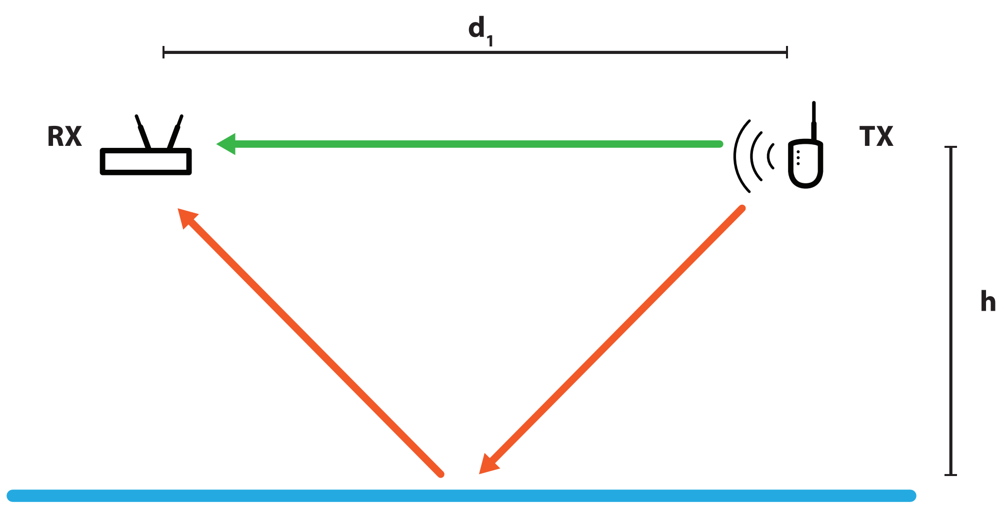


Figure 2.1.1: Two path RF propagation between a mobile transmitter (TX) and a fixed receiver (RX).

there are no walls or objects in the environment, so the ground is the only source of reflection.

The RSSI of the signal at the anchor is a function of the distance, d, between the anchor and mobile target. The average power of a sinusoid is given by

$$P = \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} (A \sin(2\pi f_0 t))^2 dt = \frac{A^2}{2},$$

where A is the amplitude of the signal and f_0 is the frequency. In the case of Figure 2.1.1, the signal at the anchor is the sum of two signals, one from the direct line-of-sight path and the other from the ground reflection path. Thus, the received signal may be written as

$$r(t) = A_\alpha \sin(2\pi f_0 t + \phi_\alpha) + A_\beta \sin(2\pi f_0 t + \phi_\beta),$$

where A_α and ϕ_α are the amplitude and phase delay of the line-of-sight signal, and A_β and ϕ_β are the amplitude and phase delay of the ground reflection. This may be rewritten as

$$r(t) = \sqrt{[A_\alpha \cos(\phi_\alpha) + A_\beta \cos(\phi_\beta)]^2 + [A_\alpha \sin(\phi_\alpha) + A_\beta \sin(\phi_\beta)]^2}$$
$$\times \sin\left(2\pi f_0 t + \tan^{-1}\left[\frac{A_\alpha sin(\phi_\alpha) + A_\beta sin(\phi_\beta)}{A_\alpha cos(\phi_\alpha) + A_\beta cos(\phi_\beta)}\right]\right).$$

RSSI is only a function of the amplitude,

$$A_r = \sqrt{[A_\alpha \cos(\phi_\alpha) + A_\beta \cos(\phi_\beta)]^2 + [A_\alpha \sin(\phi_\alpha) + A_\beta \sin(\phi_\beta)]^2}. \tag{2.1.1}$$

To compute this, the values of A_α , A_β , ϕ_α , and ϕ_β are needed, and they can be derived from Figure 2.1.1. The amplitudes are given by

$$A_\alpha = d_1^{(-n)}$$

and

$$A_\beta = d_2^{(-n)},$$

where n is the distance power law exponent, d_1 is the line-of-sight distance between the mobile target and anchor, and d_2 is the total distance of the ground reflected signal path. The distance d_1 is known, but we must compute d_2 from d_1 and the height h of the devices as

$$d_2 = 2 \times \sqrt{h^2 + \frac{(d_1)^2}{4}}.$$

The phase delays ϕ_α and ϕ_β are given by

$$\phi_\alpha = \frac{2\pi d_1}{\lambda_0}$$

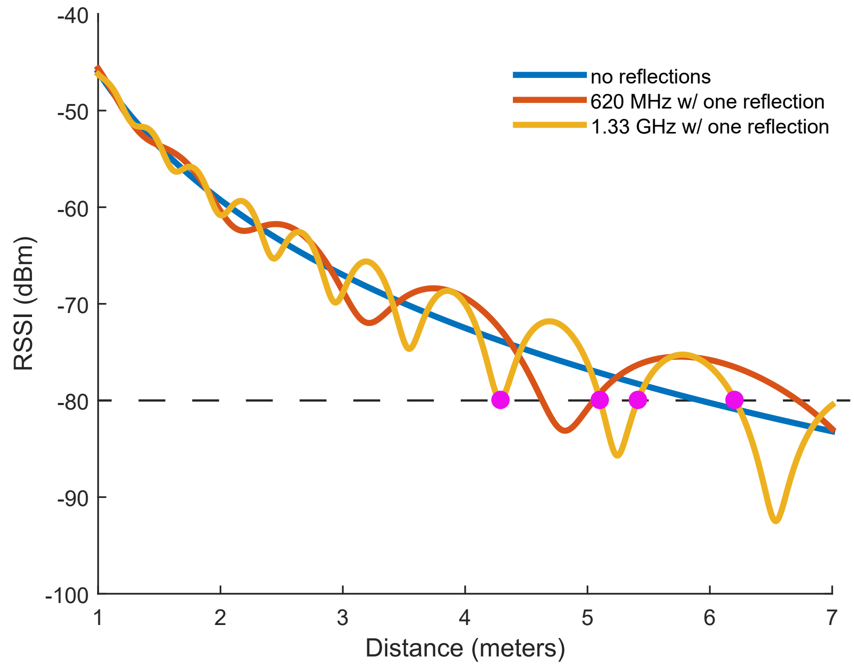


Figure 2.1.2: The plot shows RSSI as a function of distance for model in Figure 2.1.1. Two different frequencies are used.

and

$$\phi_\beta = \frac{2\pi d_2}{\lambda_0},$$

where λ_0 is the wavelength of the signal with frequency f_0. Equation 2.1.1 can now be used to compute the RSSI as a function of the distance d_1 and height h. Figure 2.1.2 shows the RSSI for three different cases. The first case, shown in blue, is the RSSI without a ground reflection. The two other cases – shown in orange and yellow – show the RSSI for f_0 equal to 620 MHz and 1.33 GHz, respectively.

Figure 2.1.2 shows that constructive and destructive interference creates local extrema at certain distances. Furthermore, it is important to observe that the occurrence of local extrema correlates with the operating frequency of the mobile target and its distance. As the frequency increases, so does the occurrence of extrema. At higher frequencies, the occurrence of the extrema is considerably higher, such that moving a small distance rapidly changes the measured RSSI. This is the fast fading effect. This is why fast fading is more prevalent in Wi-Fi enabled devices operating at 2.4 GHz than in devices operating in the 918MHz ISM band. If one samples RSSI as a function of distance with a high enough spatial resolution, one could capture the occurrence of most extrema. This captures necessary information of RSSI as a function of distance to make the process of interpolating measurements easier for any given position. It should be noted that, regardless of frequency, RSSI generally decreases with increasing distance due to the path-loss model with no reflections. In a real environment, walls and objects create multiple reflected signals contributing to the signal seen at the receiver. Researchers usually add a random component to the path-loss model to take into account the unpredictability of reflections [13, 73, 90, 108, 111, 117].

Multipath propagation introduces error in an RSSI-based localization method's location estimate by adding an element of unpredictability. For example, let us assume from Figure 2.1.2 that the RF energy exponential decay model without reflections (show by the blue line) is used to determine the location of the mobile transmitter. If the receiver determines an RSSI of -80 dBm for a received signal, then the transmitter is at a distance of six meters assuming both devices are present in a reflection free environment. However, if the environment produced a ground reflection and the signal was being transmitted at 1.33 GHz (as shown by the yellow line), the transmitter could be at four different distances: 4.3 m, 5.05 m, 5.45 m, and 6.25 m as indicated by the pink dots. The algorithm chooses one of them with a chance that it is the wrong location.

It must be noted that this example illustrates a simplified model of multipath propagation in the sense that there are only two paths. In a real indoor environment, countless number of paths exist — along side countless propagation factors — that affect the observed RSSI at a receiver. Researchers commonly approach these complicated problems with statistical methods, primarily using Rayleigh and Rician multipath propagation models [43] to represent a complicated channel envelope with Rayleigh and Rician distributions. Of the two, the Rayleigh model is the most popular because it assumes that all paths are relatively equal. That is, that there is no dominant path. This differs from the Rician model where more weight is given to the line-of-sight path.

2.2 Terminology for RSSI-Based Localization Methods

People have proposed a large variety of RSSI-based localization methods over the years. Bor et al. [18] note that, based on the different proposed taxonomies of localization techniques, there is a clear division between range-free and range-based localization. The difference between the two categories lies in the initial steps of the methods. Range-based techniques use RSSI to estimate the distance between a device with known location and a device with unknown location. On the other hand, range-free techniques exploit connectivity information between anchors to determine constraints on the location of mobile target [4].

Range-free techniques gain a great deal of information when an anchor with known location receives a signal from a target device with unknown location. This indicates that the target, of which we wish to know the location, is within the connectivity region of the anchor. The connectivity region of the anchors is the entire area where they can establish communication with another device. It is not important for range-free techniques to determine the exact location of the target because some application may not need absolute localization, but rather to have a general estimate. Because of

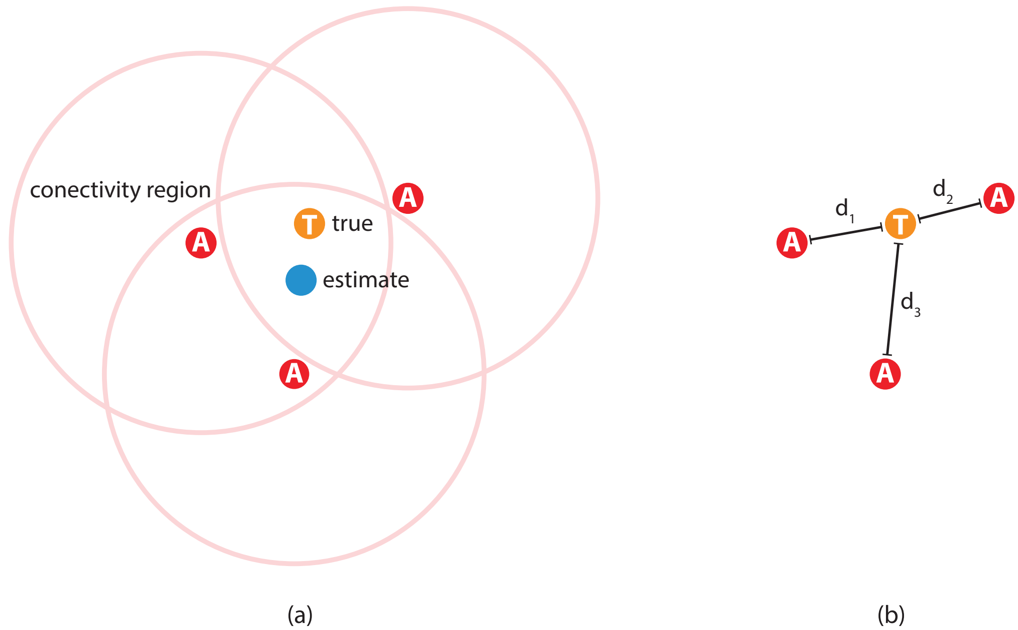


Figure 2.2.1: Sub-figure (a) illustrates range-free localization and (b) illustrates range-based localization.

this, people often choose to have low computational requirements and hardware cost at the expense of increased localization error. A simple range-free method is nearest neighbor location assignment, where the system chooses the location of the anchor that connects with the mobile target as the location estimate. This method benefits from low computational power requirements and power consumption. In the case where multiple anchors are able to hear the target, the algorithm takes additional steps to improve localization. This includes averaging the location of all the anchors receiving the signal or averaging the overlapping connectivity regions [61] as shown in Figure 2.2.1(a). The anchors are labeled with the letter A and the target is labeled with the letter T.

Range-based localization takes a different approach. Rather than simply relying on whether a target was heard or not, range-based techniques begin by using RSSI to infer the distance between the target and each of the anchors as shown in Figure

2.2.1(b). Common range-based methods include trilateration, triangulation and ring localization [23, 107, 111].

2.3 Single Channel Range-Based Localization Methods

2.3.1 Trilateration

The most popular method in RSSI-based localization is trilateration. Trilateration is a classic method for determining the location of a point using the geometry of circles or spheres. RSSI-based trilateration localization methods use RSSI to compute the distance between three or more anchors with known locations and a single mobile target with unknown location. Trilateration draws circles that have radii equal to the computed distances around the three anchors. Ideally, the three circles all intersect at one point as shown in Figure 2.3.1(a). Calculating the intersection of the circles provides the location of the target. Trilateration uses circles for 2-dimensional localization and spheres for 3-dimensional localization.

For indoor localization, there will almost never be a single point where all the circles intersect due to the seemingly noisy nature of RSSI caused by multipath propagation [118]. In some cases, as illustrated in Figure 2.3.1(b), multiple circles overlap causing uncertainty of the transmitter's location. In other instances, it may be that none of the circles intersect, as shown in Figure 2.3.1(c). The lack of a single point of intersection is the largest issue with trilateration for indoor environments. People spend much effort devising methods to solve this problem, including making artificial intersections [111].

People build all RSSI-based localization systems around the following question:

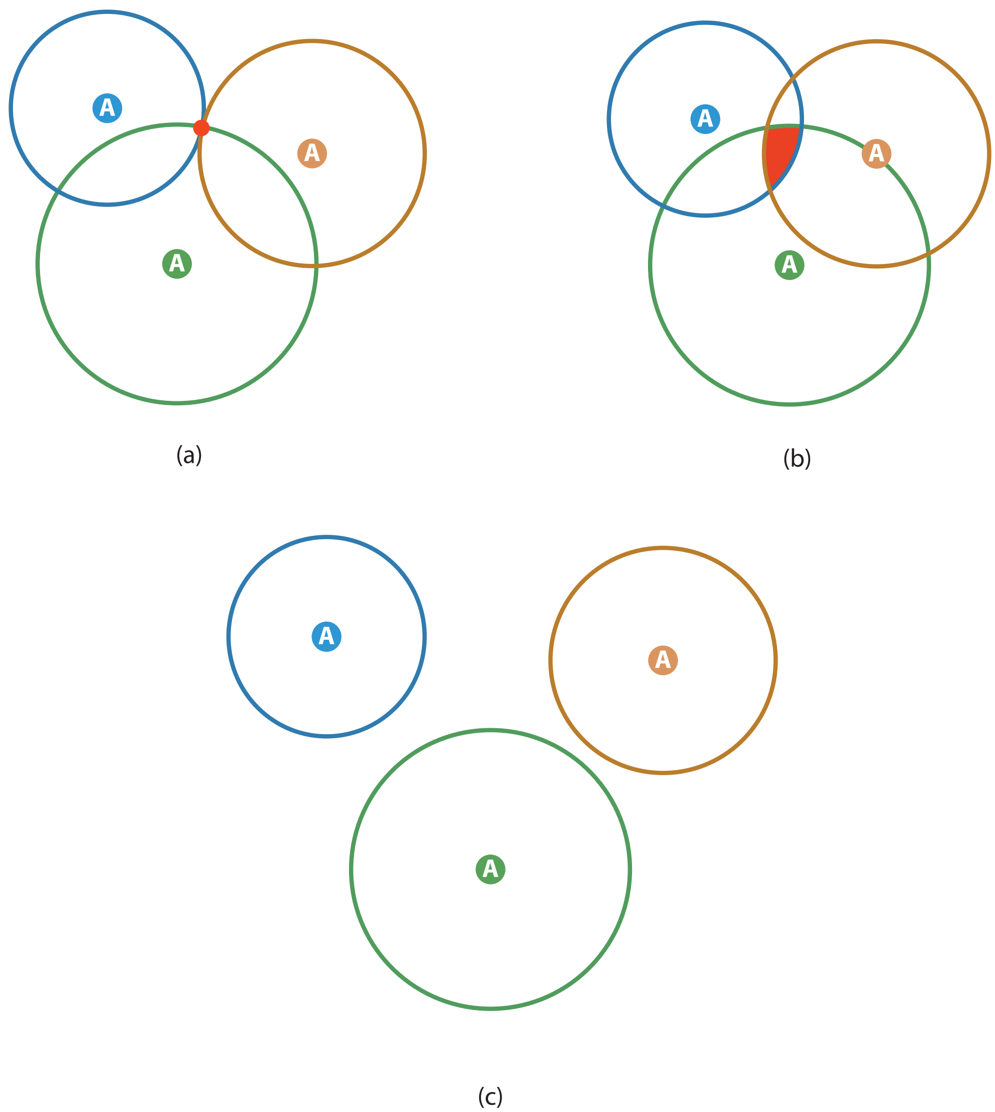


Figure 2.3.1: Sub-figure (a) has a single point of interception, (b) and (c) do not.

How can acceptable location estimates be achieved with noisy input data? Priwgharm et al. [94] use the Max-min approach, which draws squares around the circles to create smaller overlapping boxes, even if the circles do not overlap. It then chooses the center of the overlapping area as the estimated location [67]. Thaljaoui et al. [107] use a method called iRingLA. iRingLA draws rings around all circles and determines the ring thickness based on the RSSI noise in a particular environment. The algorithm then averages all points within the overlapping ring area to compute an estimated location. The method by Wang et al. [113] uses intersecting areas of circles to form points of a polygon and averages the polygon point location coordinates to provided an estimate. Researchers later improved this method into the popular weighted centroid localization (WCL) algorithm. The WCL algorithm improves accuracy by performing a weighted average on the polygon points where each polygon point is weighted by the RSSI measurements from the receivers. Liang et al. [70] use this approach for large scale WSN applications and Vari et al. [111] use WCL to investigate RSSI-based localization at the 60 GHz frequency range (IEEE 802.11ad). Others minimize and deal with noisy input data through least squares optimization which typically has higher computational requirements. The Gauss-Newton algorithm and the Lederberg-Marquette algorithm [16, 76] are examples of algorithms used for least squares optimization.

Overall, these single channel RSSI-based localization methods will have relatively large localization error. Liu et al. [71] composed a survey of wireless indoor positioning methods which is simplified in Figure 2.3.2. They concluded that single channel RSSI methods can not achieve sub-meter accuracy. Because of this relatively large localization error, the research community continues to work towards finding other alternatives.

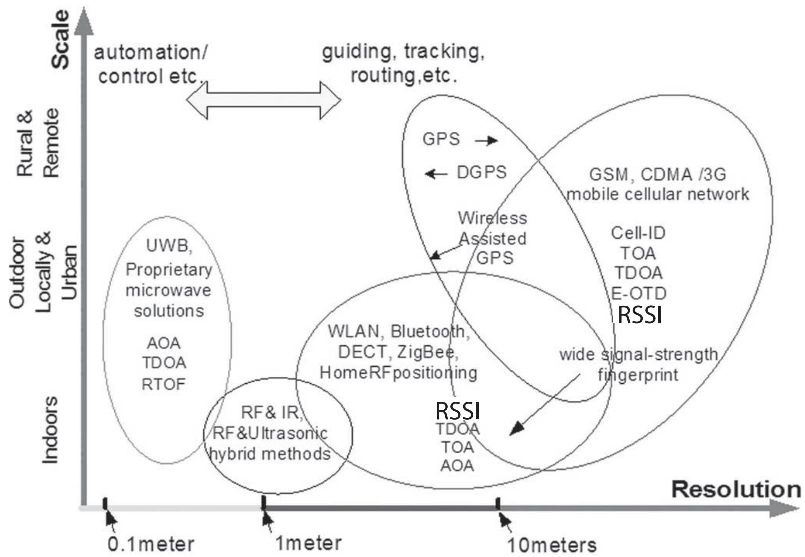


Figure 2.3.2: Outline of current wireless based positioning systems [109]

2.4 Fingerprinting for Single Channel RSSI

Pattern learning – fingerprinting – is a subcategory of range-based localization used for indoor environments [5]. People commonly use fingerprinting methods for RSSI-based localization [11, 54, 86]. It is relatively easy to distinguish fingerprinting from other methods because fingerprinting requires a calibration stage that creates a dataset by sampling RSSI from known locations, and stores them for later use. The idea is to capture a “signature” for every recorded position. In doing so, fingerprinting methods generally provide better localization results than other methods [19] at the expense of a separate calibration stage. Fingerprinting can require a great deal of time and effort to build the initial dataset, but by providing improved performance, they have captured the attention of researchers. The next section briefly covers k-nearest neighbor (kNN) and artificial neural networks (ANN) in the context of RSSI-based localization.

2.4.1 kNN

The most popular fingerprinting algorithm is the nearest neighbor approach. In nearest neighbor, the algorithm compares RSSI measurements from the mobile target to measurements captured during calibration. The algorithm computes a distance metric between all the measurements in the calibration dataset and the RSSI measurements of the mobile target. It then chooses the location associated with the closest matching measurement in the dataset as the location estimate. Bahl et al. [11] have used this approach to develop service architectures of location-aware systems to locate and track mobile users.

Algorithm developers later improved the nearest neighbor algorithm into what is known as the kNN algorithm. The kNN algorithm finds the k closest matching measurements in a dataset, where k is a specified integer, and averages the coordinates to provide a location estimate [94]. Researchers further improved the algorithm by introducing weighted averaging during the location estimation stage [41]. Fang et al. [32] performs weighted averaging with weights that are dependent on kNN distance criteria. In addition to weighted averaging, other methods have been combined with kNN. Chi et al. [24] applies the WCL algorithm after kNN to improve indoor localization for RSSI-based tracking of healthcare patients. Kasantikul et al. [56] use a particle filter after the kNN predictions which exploits a time dependent property of the measurements to improve on localization accuracy.

The distance metric is the driving mechanism of kNN; it directly determines which measurements influence the location estimate. The most common metric is Euclidean distance, but researchers use the kNN algorithm with various distance criteria including city block, Mahalanobis, and Minkowski distances [45, 103]. Guowei et al. [41] use the Jeffrey-Matusita distance formula for indoor tracking in their version of the kNN.

Some variations of the kNN algorithm assume a particular data distribution to be used as a distance metric based on the Gaussian isotropic distribution [7]. Yang et al. [116] assume a non-Gaussian distribution over their data to remove 3% of their least probable RSSI measurements. They then use weighted averaging that is dependent on the distance from the data distribution.

Fingerprinting with a kNN is a method to improve the localization error of traditional single channel localization methods. Additionally, there are other more intricate methods available to further improve localization, i.e., fingerprinting with an artificial neural network.

2.4.2 ANN

Artificial neural networks are algorithms whose inspiration comes from the biology of the human brain, neurons to be precise. Machine learning algorithm designers construct mathematical models that resemble the neural connections of a human brain working as an interconnected network [108]. Researchers have used these models in various ANN structures with different training algorithms for RSSI-based localization since the early 2000's [83] which they exploit for a variety of applications. For instance, ANN's have localized people within museums to assist in location-aware tour guide systems [110]. Battiti et al. [15] localized people within a university through a WLAN system that used a three layer feed-forward ANN trained with the one-step secant algorithm. They obtained an average localization error of 2 meters in their results. Others [10] obtained fine-grained localization (50 centimeter average localization error) to aid in indoor robotic navigation. They perform all tests in an indoor office environment and trained their three layer feed-forward ANN with the Levenberg-Marquardt algorithm. Mehmood et al. [78] cascaded several ANNs using the output of some ANNs as the inputs of other ANNs and trained all networks with genetic algorithms in order to localize a laptop within their university. Chuang et al. [25] improved localization results by providing hop count information as additional inputs and observed a 5 meter average location error during simulations. More recently, researchers used a feed-forward ANN trained with their own feature selection backpropagation artificial immune system (FSBP-AIS) algorithm to track workers in a warehouse [60]. Their training algorithm performed better than traditional backpropagation algorithms due to that fact that their FSBP-AIS model does not tend to converge towards local minima.

More recently, the radial basis function neural network (RBFNN) structure has become more popular for indoor RSSI-based localization with ANNs; especially after 2012. RBFNNs are a special class of ANN where some layers consist of Gaussian kernels. A different class of training algorithms are used for these networks. Typically, training is divided into two stages: first, the center and widths of the Gaussian kernels are determined and then the network learns all other parameters [108]. Carlson et al. [22] used a (RBFNN) on localization data to monitor the health of the elderly. Their model was trained using linear optimization and later their localization estimates were refined by using a Viterbi algorithm. Goa et al. [40] used the difference of RSSI as additional inputs to their RBFNN which they trained with a fuzzy clustering algorithm. Others combined the RBFNN and Particle filter to improve localization results [82]. Their RBFNN provided a real-time location estimate and the particle filter was used to predict the next location.

In summary, single channel RSSI-based localization is going to be problematic due to localization error coming from multipath propagation. Fingerprinting with the ANN and kNN can help address this issue, but even though using these methods render better localization results than traditional methods, the results will still have a relatively large localization error.

2.5 Multichannel RSSI Methods

As discussed in section 1.2.1, a commonly used indoor line-of-sight path-loss model is

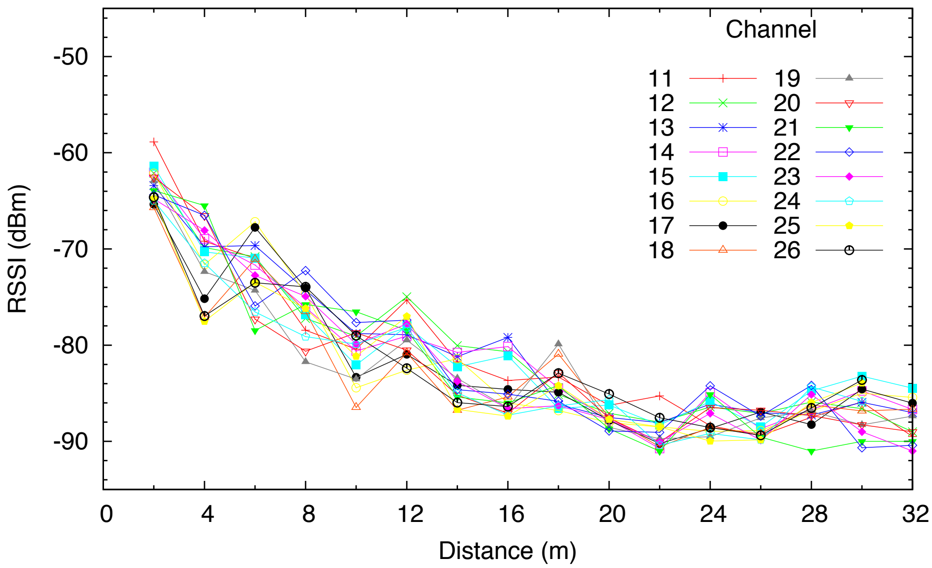
$$\mathbf{RSSI}(d) = \mathbf{RSSI}_0 - 10 n_p \log \frac{d}{d_0} + X_\sigma, \tag{2.5.1}$$

where the random variable X_σ represents the erratic behavior of RSSI due to multipath propagation. Bor et al. [18] recorded RSSI on 16 different channels for IEEE 802.11 compliant devices transmitting at various distances and their data is consistent with the path-loss model of equation 2.5.1. Their data, shown in Figure 2.5.1(a), shows that the RSSI drastically changes over increments as small as two feet. This graph also shows that RSSI is different for the same location when measured on different channels.

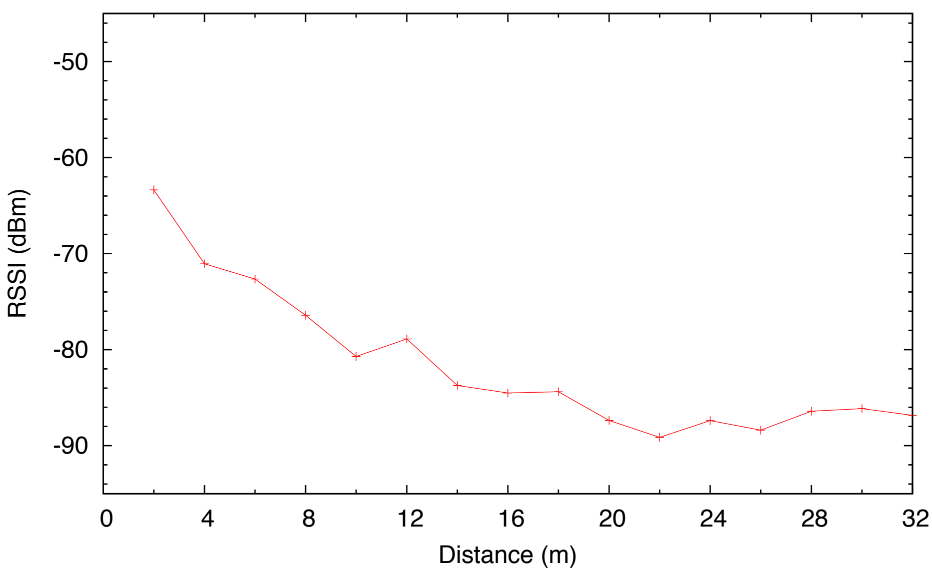
The RSSI for a single channel varies rapidly over small distances, but if all the measurements for a given location are averaged over all the channels, the results approximate the path-loss model without the random component. Figure 2.5.1(b) shows the results when the measurements in Figure 2.5.1(a) are averaged over all channels for each location. Doing this mitigates multipath propagation effects [115]. Because of this, many researchers apply channel averaging to improve localization accuracy in RSSI-based systems. Many other frequency diversity methods do exist, but frequency averaging is the most popular. Bardella et al. [13] use IEEE 802.15.4 compliant devices operating at 2.4 GHz to measure RSSI from the 16 defined channels. They show that localization accuracy is improved by averaging measurements that are collected on different channels. Using the same standard, Ladha et al. [66] were able to significantly reduce the average root mean squared error for estimating a device's location within an office environment. Pricone et al. [93] used a system that averaged RSSI over four channels to locate basketball players in a gym. They used Memsic IRIS anchors operating in the 2.4 GHz ISM band with their own TDMA communication protocol designed for channel hopping.

2.5.1 Other Multi-Frequency Approaches

Although RSSI averaging is the most popular method for improving localization results when using multi-frequency data, other methods exist. Fink et al. [33] use the weighted centroid localization (WCL) algorithm without averaging data from multiple channels with stationary nodes transmitting at two frequencies. The dynamic sensors, or target sensors to be located which are referred to as BN, have two antennas placed in different locations of the board, one for each frequency. They use a total of four antennas for each target sensor. By doing so they achieve frequency diversity and spatial diversity at the same time. Figure 2.5.2 shows the transmitting (BN) and receiving (RN) devices communicating to each other. They are able to obtain four different RSSI measurements, each with a different frequency and spatial offset for the same sensor location. Their algorithm starts by converting received RSSI measurements into weights for each RN. Then, an adaptive WCL algorithm estimates the sensor location by using a modified weighted average approach. To improve accuracy, Fink et al. also uses a plausibility filter where movement restriction is enforced. This limits the maximum distance that a BN may travel during the prediction stage and thus lowers error from predicting unrealistic movement by imposing a maximum velocity that a BN can travel. Fink et al. improves upon his method in later work by refining localization results through data fusion [34]. A six axis inertial measurement



(a) Mean RSSI versus distance per channel.



(b) Mean RSSI versus distance over all channels.

Figure 2.5.1: Graph (a) shows RSSI as a function of distance for multiple communication channels. Graph (b) shows the data in graph (a) averaged over all channels.

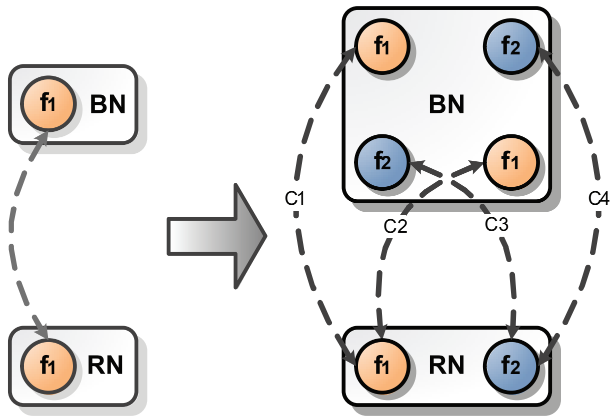


Figure 2.5.2: A diagram showing how Fink's et al. devices communicate to each other.

unit (IMU) provides an additional location prediction method through the use of a Kalman filter. A final filter combines the prediction from the two methods, one from WCL and another from the Kalman method, to provide an updated estimate. These methods may offer an alternative to channel averaging based localization, but their systems add additional hardware to the system which increases the complexity and drives up cost.

Another method of localizing a target while operating on multiple frequencies is radio interferometric positioning system (RIPS). Localization by RIPS is achieved by transmitting two RF signals with slightly different frequencies. The composite signal at the receiver's side will have a low frequency envelope such that neighboring devices can measure its power with less expensive hardware than that of measuring time-of-arrival. Figure 2.5.3 displays the computation of the phase offset δ between C and D used to compute an AOA measurement. The method then infers location from AOA measurements with a variety off-the-shelf algorithms. Maroti et al. [75] first introduced RIPS for 3D positioning of wireless sensor networks. They later improved their method through various developments [63, 64, 65]. It is worth noting that RIPS is not indoor localization method and was only presented because it uses multiple

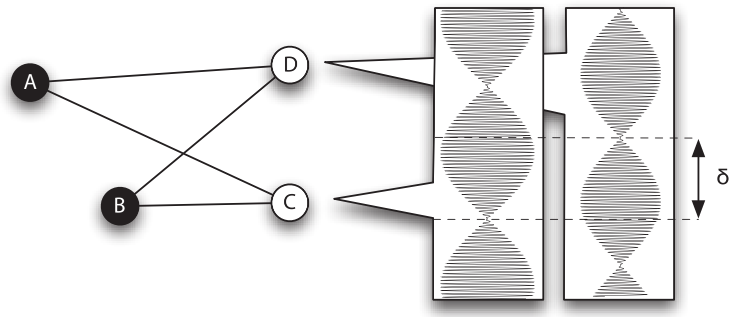


Figure 2.5.3: An illustration of the radio interferometric ranging technique.

frequencies in it's localization scheme. Neither does it use RSSI. This work looks to use RSSI for localization because it relatively easy to have access too. Even if RIPS was deployable indoors, the complexity of the system would drive up the cost.

2.5.2 Fingerprinting with Multichannel Data

In one instance, found in the work of Bor et al. [18], a machine learning algorithm uses multiple frequencies for indoor localization where a nearest neighbor algorithm was used along side RSSI fingerprinting. During a training stage, Bor et al. collected RSSI data in an office environment for different locations to create a dataset that would later be used as a look-up table. The nearest neighbor algorithm searches the dataset to find the closest matching data point with a new and unknown RSSI measurement during the prediction stage. The location of the closest matching data point is then the predicted location. Additionally, the simple algorithm's prediction accuracy increases when averaging measurements over multiple channels. This method does not allow for fine grain localization but it works well for room level localization. What Bor et al. learned with these results is that fingerprinting with a nearest neighbor algorithm can improve localization error. If this simple method can improve results, then it would be expected that more complicated algorithms like the kNN can surpass its performance.

2.6 Summary

The presented literature shows that multichannel RSSI can help mitigate localization error. This is beneficial because it is relatively easy to measure RSSI from multiple channels. Currently, most work on multichannel localization averages multichannel RSSI. Instead of using channel averaging, this work focuses on treating multichannel RSSI separately and combining multichannel fingerprinting with algorithms that include k-nearest neighbor and artificial neural networks.

Chapter 3

Fingerprinting Methods for Localization

This chapter introduces fingerprinting methods that use frequency diversity to mitigate localization error. As stated in Chapter 1, the hypothesis is that a multichannel RSSI fingerprint of the environment is capable of providing more information regarding a mobile target's location than a single RSSI measurement. To test this hypothesis, the performance of various fingerprinting algorithms are evaluated based on their ability to estimate the mobile target's location.

This chapter introduces three methods for RSSI fingerprinting. The first is a k-nearest neighbor (kNN) implementation that stores calibration data and later uses it as a look-up table to interpolate an active tag's position using weighted averaging. The second uses a data driven Neural Network model. The third method uses statistical modeling and particle filtering to maximize the *a posteriori* probability of a current location estimate.

For 2-dimensional localization using multichannel RSSI, let the 2-dimensional location of a mobile target be

$$\mathbf{s}_m = [\ s_{m,x}, \quad s_{m,y}\ ],$$

where $s_{m,x}$ denotes the first spatial coordinate and $s_{m,y}$ denotes the second coordinate. The associated RSSI measurement recorded for multiple communication channels at location $\mathbf{s}_m$ is

$$\mathbf{z}_m = [\ z_{m,1}, \quad z_{m,2}, \quad \cdots, \quad z_{m,C}\ ],$$

where C denotes the total number of channels used.

The goal of the localization methods is to calculate the true location $\mathbf{s}_m$ from the measurement $\mathbf{z}_m$. In order to do so, the fingerprinting algorithms must establish a relationship between $\mathbf{s}_m$ and $\mathbf{z}_m$ from a calibration dataset prior to operation. The locations in the calibration dataset, or *training dataset*, are

$$\mathbf{s}_{1:M} = \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \vdots \\ \mathbf{s}_M \end{bmatrix},$$

where M is the total number of positions in the training dataset. The corresponding RSSI at each set of positions $\mathbf{s}_{1:M}$ is

$$\mathbf{z}_{1:M} = \begin{bmatrix} z_{1,1}, & z_{1,2}, & \cdots, & z_{1,C} \\ z_{2,1}, & z_{2,2}, & \cdots, & z_{2,C} \\ \vdots, & & & \\ z_{M,1}, & z_{M,2}, & \cdots, & z_{M,C} \end{bmatrix}.$$

In order to evaluate an algorithm's performance and to account for the possibility of overfitting, a separate dataset is required for testing that provides unseen data measurements to evaluate the robustness and accuracy of localization. The *testing dataset* also consists of 2-dimensional location and RSSI measurement data given by $\mathbf{s}_{1:N}$ and $\mathbf{z}_{1:N}$, respectively, where N denotes the total number of measurements in the testing dataset. For this work, both the testing and training dataset come from a single dataset collected through the same experimental procedure. The entire dataset is then split into an 80:20 ratio: 80% of the data is used for the training dataset while the remaining 20% is used for testing dataset which is a common ratio among the machine learning community. This ensures that both datasets are sampled from the same environment with the same sampling probability distribution, while still being independent of each other in order to avoid the problem of overfitting. More details on the data collection process are provided in Chapter 4.

To be clear, the index m and constant M will be used exclusively for the training dataset while the index n and N will be used exclusively for the testing dataset.

3.1 k-Nearest Neighbors Algorithm

The k-nearest neighbor (kNN) algorithm, one of the simplest fingerprinting algorithms, was proposed in the 1960's and is still commonly used today [27]. The intuition behind the algorithm is simple: when an application requires a prediction for an unseen data sample, the kNN algorithm searches through the training dataset for the k-most similar samples [20]. The algorithm then uses the prediction attributes of the most similar samples to compute the estimate for the unseen sample. Many researchers have demonstrated that kNN is computationally efficient for many applications with acceptable accuracy [69], especially in clustering, classification, and

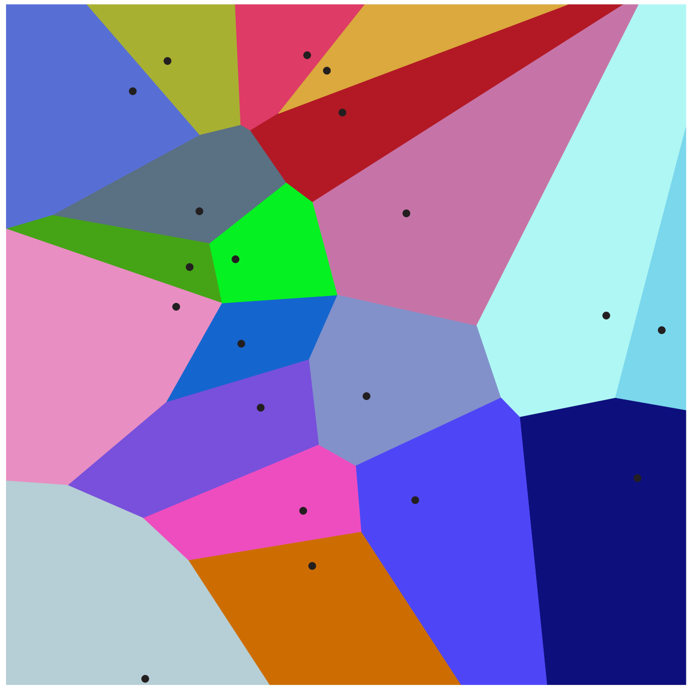


Figure 3.1.1: Voronoi diagram

regression. It can be used for interpolation, as illustrated by the Voronoi diagram shown in Figure 3.1.1. Here, sub-spaces are divided on a 2-dimensional plane based on distances, where the different subareas are shaded in assorted colors corresponding to the closest samples on the plane. This figure provides a visual example of piece-wise constant interpolation using a single nearest neighbor algorithm.

In the context of RSSI-based localization, kNN provides a location estimate $\hat{\mathbf{s}}_n$ using only the RSSI measurement $\mathbf{z}_n$ and a database of known location RSSI pairs $(\mathbf{s}_{1:M},\mathbf{z}_{1:M})$. The index n denotes the unseen measurement of interest. The algorithm works as follows:

- Compute the distance d between each new measurement $\mathbf{z}_n$ and all known measurements in the training dataset $\mathbf{z}_{1:M}$.

- Select the k neighbors within the training dataset $\mathbf{z}_{1:M}$ with the smallest distances.

- Compute $\hat{\mathbf{s}}_n$ as a weighted average of all known measurements in $\mathbf{s}_{1:M}$ corresponding to the k-nearest neighbors of $\mathbf{z}_n$ within $\mathbf{z}_{1:M}$ using

$$\hat{\mathbf{s}}_n = \frac{\sum_{i=1}^{k} d_i \mathbf{s}_i}{\sum_{i=1}^{k} d_i}.$$

- Repeat the previous three steps for all unseen measurements.

- Stop when $\hat{\mathbf{s}}_N$ is computed.

Various distance criteria have been used for the kNN algorithm including the Manhattan and Minkowski distances [102]. The majority of researchers use the Euclidean distance because this metric is also often considered as the standard choice when no prior knowledge is available about the data's distribution [114]. For this reason, this work used the Euclidean distance given by

$$d(\mathbf{z}_n, \mathbf{z}_m) = ||\mathbf{z}_n - \mathbf{z}_m||,$$

where $\mathbf{z}_n$ and $\mathbf{z}_m$ are RSSI measurements in dBm for the testing and training dataset, respectively, and

$$||\mathbf{z}_n - \mathbf{z}_m|| = \sqrt{(z_{n,1} - z_{m,1})^2 + (z_{n,2} - z_{m,2})^2 + \cdots + (z_{n,C} - z_{m,C})^2},$$

where $z_{n,c}$ and $z_{m,c}$ are RSSI measurements for channel c. The algorithm calculates the Euclidean distances between every new measurement and all of the current measurements in the training datasets during the first step. The distances are then compared to each other to select the measurements with the smallest distance. That determines which measurements will be used to compute a weighted average for the location estimate.

3.2 Artificial Neural Networks

Artificial neural networks (ANN) are used for state-of-the-art machine learning frameworks [1, 26, 53] and were inspired by the biological structure of neural networks in the human brain. Human neurons interconnect in large intricate networks to transfer information amongst each other with electrochemical signals to produce thoughts and actions. A neuron can be simplified into dendrites, axons, and a nucleus. The dendrites and axons analogize as the inputs and outputs of each neuron. Figure 3.2.1 displays a simplified image of two interconnected neurons. The axons (outputs) of the first neuron transfer electrochemical signals to the second neuron's dendrites (inputs). The receiving neuron processes these signals within its nucleus to either produce a response (or not) and then transmit its own output signal, through its axons, to other neurons.

Early neural network developers conceived the artificial neuron with the concept of a biological neural network architecture. The crude analogy between artificial neurons and the biological neuron is that the connections between nodes represents the axons and dendrites, the connection weights represent the synapses, and the activation function approximates the activity in the soma [52]. Figure 3.2.2 illustrates an artificial neuron and shows multiple inputs, an output, and an activation function analogous to a biological neuron's soma. The activation function is what produces a neuron's output which is dependent on its input and the selected activation function. In the context of multichannel RSSI fingerprinting, the input A to the activation

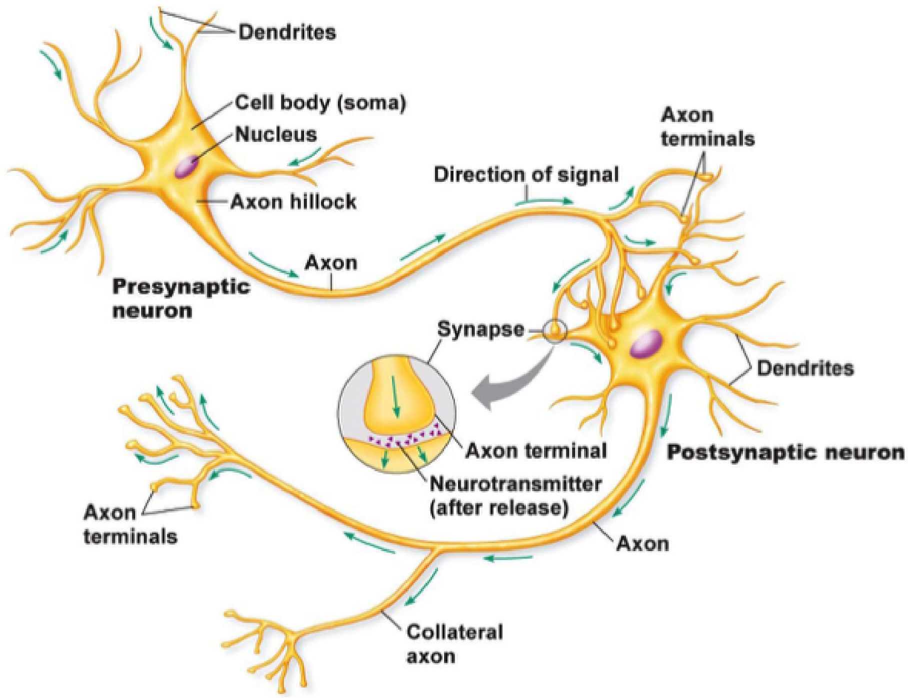


Figure 3.2.1: Interconnected human neurons [31]

function is

$$A = w_1 z_1 + w_1 z_2 + \cdots + w_C z_C,$$

where $z_{1:C}$ are the input data and $w_{1:C}$ are the corresponding weights. The output of the activation function is called the activation a. Both the biological network and the ANN learn by incrementally adjusting the magnitudes of their weights or synapses [120].

Certain activation functions can introduce nonlinearity in the network. Without these functions, the network can only learn functions that are linear combinations of the inputs. Gaussian, step, threshold, sigmoid, and rectified linear units are examples of such functions. This work uses the sigmoid function because it possesses the distinctive properties of continuity and differentiability on the interval $(-\infty, \infty)$, which are both essential requirements in back-propagation learning [14]. The sigmoid

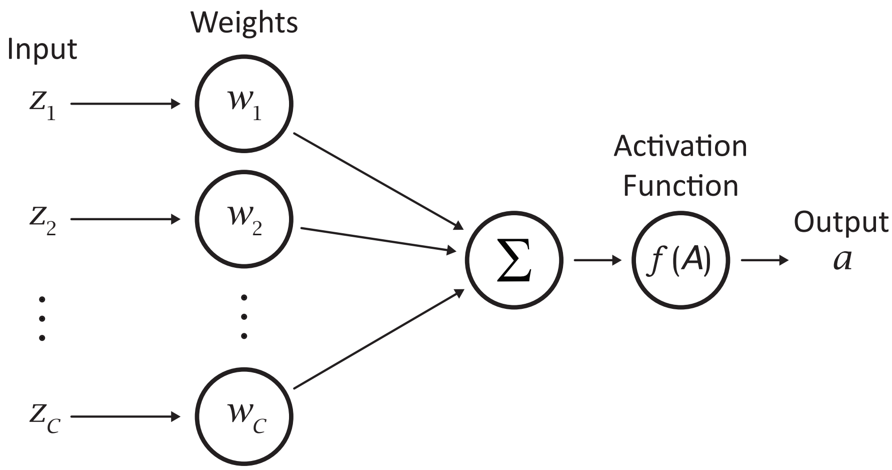


Figure 3.2.2: The artificial neuron

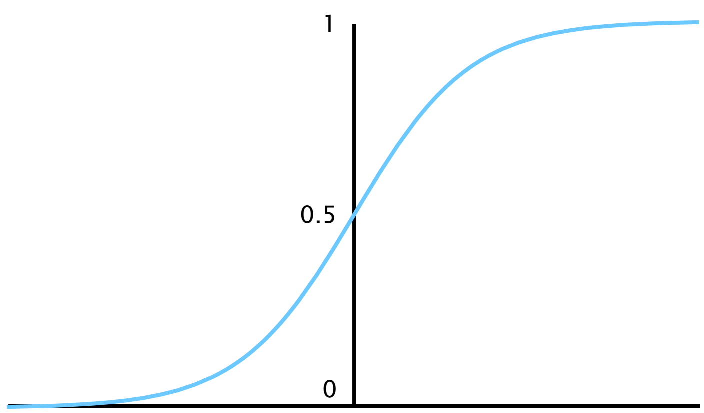


Figure 3.2.3: The sigmoid activation function

function is

$$f(A) = \frac{1}{1 + e^{-\beta A}},$$

where β is a constant that determines the width of the sigmoidal shape. Low input values (far into the negatives) produce an output close to zero; high input values result in an output close to one. The sigmoid function's response is shown in Figure 3.2.3.

ANNs are typically modeled as collections of neurons that are interconnected in acyclic graphs [55]. In other words, the outputs of some neurons can become inputs to other neurons. The algorithm propagates input data through a network from start to finish in a process which is referred to as a *forward pass*. Additionally, ANN models are often organized into distinct layers of neurons instead of amorphous blobs of interconnected neurons. For regular neural networks, the most common layer type is the *fully-connected layer* in which neurons between two adjacent layers are fully pairwise connected, but neurons within a single layer share no connections [55]. Figure 3.2.4 illustrates a three layer feed-forward ANN using a stack of fully connected layers and also shows the direction of data flow. The first layer represents the input data, the intermediate layer is called the *hidden layer*, and last layer is the output of the ANN. With an exception to the input layer, each layer has a *bias* value as an input to neurons that introduces a bias offset. Figure 3.2.4 depicts the bias values as the circles labeled with b. The bias allows the algorithm to modify the bias weight value to shift a neuron's response either the left or the right. This may be necessary to learn certain features between the training and testing datasets.

3.2.1 Training with the Back-Propagation Algorithm

ANNs undergo training to learn relationships between input and output data through adjusting network weights and biases. The network's ultimate goal is to make predictions on a testing dataset with the least amount of error so that the algorithms can be deployed to solve real problems. Researchers have used various training methods of various complexity including Levenberg-Marquardt, adaptive sub-gradient, and even a Kalman filter [8, 10, 59].

This work trains an ANN with Levenberg-Marquardt backpropagation because it

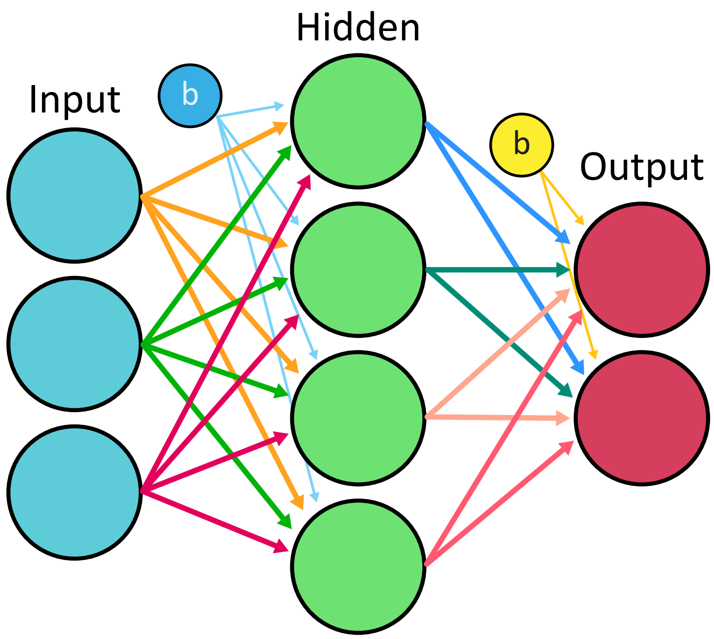


Figure 3.2.4: A three-layer feed-forward perceptron neural network

is a very efficient algorithm for networks with less than a few hundred weights even when compared with conjugate gradient techniques [42]. The Levenberg-Marquardt backpropagation algorithm is well documented in the work of Hagan et al. [42]. Before explaining the Levenberg-Marquardt algorithm, the gradient descent algorithm is first presented since Levenberg-Marquardt builds upon gradient descent to create a more efficient training algorithm.

In gradient descent, as Jia et al. [53] describe, ANN layers have two key responsibilities for the operation of the network as a whole. A *forward pass* that takes the inputs and produces the outputs at the final layer, and a *backwards pass* that takes the gradients with respect to the output, and computes the gradients with respect to the parameters and to the inputs, which are in turn back-propagated to earlier layers. This process ultimately updates all weights and bias values. The forward pass begins by considering a multilayer neural network where the net input A to a neuron j in layer $l+1$ is

$$A_{l+1}(j) = \sum_{i=1}^{H_l} w_{l+1}(i,j)a_l(i) + b_{l+1}(j),$$

where H_l is the total number of neurons in layer l. The activation of neuron j is

$$a_{l+1}(j) = f_{l+1}(A_{l+1}(j)).$$

For an L layer network, the algorithms vectorize the previous expression for all network layers into a system of equations given by

$$\mathbf{a}_0 = \mathbf{z} \tag{3.2.1}$$

and

$$\mathbf{a}_{l+1} = f_{l+1}(\mathbf{w}_{l+1}\mathbf{a}_l + \mathbf{b}_{l+1}), \qquad l = 0, 1, \cdots, L-1 \ . \tag{3.2.2}$$

These equations make up the forward propagation stage. Note that the task of the ANN is to learn associations between a specified set of input-output pairs

$$\{(\mathbf{z}_1, \mathbf{s}_1), (\mathbf{z}_2, \mathbf{s}_2), \cdots, (\mathbf{z}_M, \mathbf{s}_M)\},$$

so the performance index (also known as the cost function or the objective function) for the mth input of the ANN is given by

$$E = \frac{1}{2}\mathbf{e}_m^\top \mathbf{e}_m,$$

where $\mathbf{e}_m = \mathbf{s}_m - \mathbf{a}_{L,m}$ is the prediction error for the mth input, $\mathbf{z}_l$, and $\mathbf{e}_m^\top \mathbf{e}_m$ is the squared error. Using this, the standard backpropagation algorithm uses an approximate steepest descent rule. Since the observation of convergence towards a local minima always moves towards the negative gradient of the convex function E [2], the approximate steepest (gradient) descent algorithm is then

$$\Delta w_l(h, c) = -\alpha \frac{\partial E}{\partial w_l(i, j)} \tag{3.2.3}$$

and

$$\Delta b_l(h) = -\alpha \frac{\partial E}{\partial b_l(i)} \tag{3.2.4}$$

where α is the learning rate. The algorithm obtains the gradients through

$$\frac{\partial E}{\partial w_l(i, j)} = \xi_l(i) a_{l-1}(j) \tag{3.2.5}$$

and

$$\frac{\partial E}{\partial b_l(i)} = \xi_l(i), \tag{3.2.6}$$

where $\xi_l(i)$ is the sensitivity of the cost function to changes in the net input A of neuron i in layer l. The algorithm exploits the fact that the sensitivities satisfy the following recurrence relation

$$\xi_l = \mathbf{f}_l(\mathbf{A}_l)\mathbf{w}_{l+1}^\top \xi_{l+1}, \tag{3.2.7}$$

where it is initialized at the final layer as

$$\xi_L = -\mathbf{f}_L(\mathbf{A}_L)(\mathbf{s}_m - \mathbf{a}_m). \tag{3.2.8}$$

In summary, the algorithm first performs a forward pass using Equations 3.2.1 and 3.2.2; then it performs a backwards pass using Equations 3.2.8 and 3.2.7; and finally updates the weights and biases using Equations 3.2.3, 3.2.4, 3.2.5, and 3.2.6.

3.2.2 Training with the Levenberg-Marquardt Algorithm

The Levenberg-Marquardt algorithm is a balancing act between an approximation to Newton's optimization method and an approximation to the gradient descent rule [42]. Suppose a problem seeks to minimize a function with respect to the parameter vector $\mathbf{w}$, then Newton's method would be

$$\Delta\mathbf{w} = [J^\top(\mathbf{w})J(\mathbf{w})]^{-1}J^\top(\mathbf{w})\mathbf{e}(\mathbf{w}),$$

where $J(\mathbf{w})$ is the Jacobian matrix and $e(\mathbf{w})$ is the error vector. The Levenberg-Marquardt modification [42] to the Gauss-Newton method is

$$\Delta\mathbf{w} = [J^\top(\mathbf{w})J(\mathbf{w}) + \mu I]^{-1}J^\top(\mathbf{w})\mathbf{e}(\mathbf{w}). \tag{3.2.9}$$

By varying the combination coefficient μ, the algorithm performs parameter updates by adaptively changing between the gradient descent rule and Gauss-Newton update [36]. The coefficient μ is determined by the following rule

$$\mu = \begin{cases} E(w_t) > E(w_{t-1}), & \beta\mu_0 \\ E(w_t) \leq E(w_{t-1}), & \frac{\mu_0}{\beta} \end{cases},$$

where the coefficient μ_0 is multiplied by some value β whenever a step would result in an increase of the cost function $E(w)$. When a step reduces $E(w)$, μ_0 is divided by β. If μ is small, then the method approximates the Gauss-Newton method; if it is large, it approximates the gradient descent rule. It is a disadvantage to always use the Gauss-Newton method throughout training since the search space is only convex around the point of interest and has multiple local minima. Using the "optimal" step from the Gauss-Newton method would most likely guarantee a divergence from the point of interest due to the fact that the objective function is not globally convex. Using a variable step size allows for faster convergence when approaching a minima (undergoing a constant negative slope) while still maintaining stability. Knowing this, the last modification to the standard backpropagation algorithm is seen at the final layer as

$$\Delta_L = -\mathbf{f}_L(\mathbf{A}_L), \tag{3.2.10}$$

where each column of the matrix Δ_L is a sensitivity vector that must be backpropagated through the network to produce one row of the Jacobian matrix [42].

In summary, the algorithm first performs a forward pass to compute the prediction error at the output of the network. Then the algorithm computes the Jacobian matrix using Equations 3.2.5, 3.2.6, 3.2.7, and 3.2.10. Finally, it solves for $\Delta\mathbf{w}$ using equation 3.2.9.

3.3 The Particle Filter

Since their introduction in 1993, particle filters have become a popular class of estimators for nonlinear non-Gaussian problems [30]. This filtering technique handles situations where information about a random process is desired. More formally, particle filtering is a general Monte Carlo (sampling) method that performs inference of state-space models where the state of a system evolves in time and collects information via noisy measurements made at each time step [84].

The filter is based on Bayesian principles that have provided a rigorous general framework for dynamic state estimation problems [51]. Bayesian tracking methods typically construct probability density functions (PDF) for states based solely on all previous observations [38]. Although these methods work particularly well for lin-

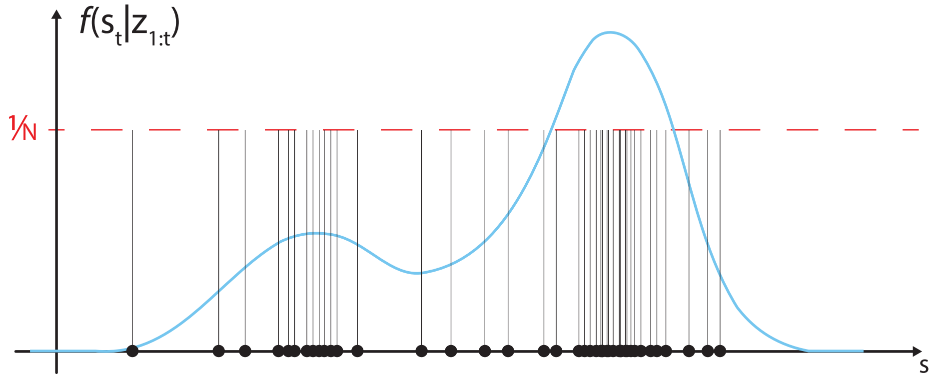


Figure 3.3.1: Particles approximate a non-Gaussian distribution

ear Gaussian (LG) estimation problems, there are no general analytic (closed form) expressions for the required PDF of nonlinear non-Gaussian (NLNG) estimation problems. Particle filters cleverly approach NLNG problems by representing the required posterior PDF as a set of random samples (particles) with associated weights, then computing estimates based on these samples and weights [100]. Figure 3.3.1 displays a set of particles that approximate a NLNG distribution (blue line). Particle filters have the ability to approximate any arbitrary PDF with a relatively large set of particles, making the algorithm a suitable choice for problems with non-Gaussian, multi-modal PDFs.

This section gives a general overview of the particle filter algorithm. For a thorough understanding of the algorithm, refer to the work by Arulampalam et al. [9] which includes a tutorial of the particle filter. In order to formulate the particle filter problem, the state transition model and the measurement equation must first be defined. The state transition model, otherwise known as the discrete-time state-space model, and the measurement equation are

$$\mathbf{s}_t = f_t(\mathbf{s}_{t-1}, \eta_{t-1})$$

and

$$\mathbf{z}_t = g_t(\mathbf{s}_t, \nu_t),$$

respectively, where $\mathbf{s}_t$ represents the state of the system at time t and $\mathbf{s}_{t-1}$ represents the previous state. Also, $\mathbf{z}_t$ is a noisy measurement which is the only information obtained from $\mathbf{s}_t$. The function f_t describes the evolution of the state and g_t is the function describing the measurement process. Both f_t and g_t are possibly nonlinear and time-dependent functions. η_{t-1} and ν_t are the state and measurement noise, respectively.

The particle filtering problem involves computing the estimate of the state $\mathbf{s}_t$ at time t given all measurements up to and including t (also written as $\mathbf{z}_{1:t}$). The Bayesian solution must first be formalized because the particle filter builds upon Bayesian principals.

3.3.1 The Optimal Bayesian Tracking Solution

In a Bayesian setting, the optimal Bayesian solution estimates the current state $\mathbf{s}_t$ by computing the PDF $p(\mathbf{s}_t|\mathbf{z}_{1:t})$ of the current state using all previously known observations. The computation process performs this in two steps: the prediction step and the update step. In the prediction step, $p(\mathbf{s}_t|\mathbf{z}_{1:t-1})$ is

$$p(\mathbf{s}_t|\mathbf{z}_{1:t-1}) = \int_{-\infty}^{\infty} p(\mathbf{s}_t|\mathbf{s}_{t-1})p(\mathbf{s}_{t-1}|\mathbf{z}_{1:t-1})d\mathbf{s}_{t-1}, \tag{3.3.1}$$

where $p(\mathbf{s}_{t-1}|\mathbf{z}_{1:t-1})$ and $p(\mathbf{s}_t|\mathbf{s}_{t-1})$ are both assumed to be known and $p(\mathbf{s}_t|\mathbf{s}_{t-1})$ is given by the state transition model. At this point, $p(\mathbf{s}_t|\mathbf{z}_{1:t-1})$ is the prior estimate of the state before receiving the measurement at time t. When a new measurement is made, then one may proceed to the update step by using Bayes' rule to obtain the posterior PDF

$$p(\mathbf{s}_t|\mathbf{z}_{1:t}) \propto p(\mathbf{z}_t|\mathbf{s}_t)p(\mathbf{s}_t|\mathbf{z}_{1:t-1}), \tag{3.3.2}$$

where $p(\mathbf{z}_t|\mathbf{s}_t)$ is the distribution of z_t with the newly available measurement.

Equations 3.3.1 and 3.3.2 have a recurrence relation and form the basis for the optimal Bayesian solution. In general, the recursive propagation of the posterior PDF cannot be computed analytically and can only be done in a restrictive set of cases. But, when the analytical solution is intractable, e.g. the NLNG case, particle filters can provide an approximation to the optimal Bayesian solution.

3.3.2 Sequential Monte Carlo Simulation

A sequential Monte Carlo (SMC) simulation is the most basic method used to approximate the optimal Bayesian solution. The simulation produces random samples with associated weights to represent the required posterior density $p(\mathbf{s}_t|\mathbf{z}_{1:t})$. These random samples are known as particles and the representation is

$$p(\mathbf{s}_t|\mathbf{z}_{1:t}) \approx \sum_{i=1}^{N} \omega_t^i \delta(\mathbf{s}_t - \mathbf{s}_t^i), \tag{3.3.3}$$

where ω_t^i are the associated particle weights, i denotes the particle index, N is the total number of particles, and $\delta(\cdot)$ denotes the Dirac delta function. As the number of particles increases, the accuracy of the approximation improves. The weights are then chosen using the principle of importance sampling [29].

3.3.3 Sequential Importance Sampling

It is common to see the sequential importance sampling (SIS) particle filter and SMC as being presented as the same thing in much of the literature [30], but in fact the

Algorithm 1 SIS Particle Filter [9]

$[\{\mathbf{s}_t^i, \omega_t^i\}_{i=1}^N] = \texttt{SIS}[\{\mathbf{s}_{t-1}^i, \omega_{t-1}^i\}_{j=1}^N, \mathbf{z}_t]$

- FOR $i = 1 : N$
  - Draw $\mathbf{s}_t^i \sim q(\mathbf{s}_t^i | \mathbf{s}_{t-1}^i, \mathbf{z}_t)$
  - Assign the particle a weight, ω_t^i, according to equation 3.3.4
- END FOR

SIS algorithm is a Monte Carlo (MC) method that forms the basis for the particle filter. For a problem where it is difficult to draw particles from $p(\mathbf{s}_{0:t}|\mathbf{z}_{1:t})$ for equation 3.3.3, one could draw particles from the importance PDF $q(\cdot)$ that is related to the particles weights ω_t^i by

$$\omega_t^i \propto \frac{p(\mathbf{s}_{0:t}^i|\mathbf{z}_{1:t})}{q(\mathbf{s}_{0:t}^i|\mathbf{z}_{1:t})}.$$

This relationship is rewritten[9] as

$$\omega_t^i \propto \omega_{t-1}^i \frac{p(\mathbf{z}_t|\mathbf{s}_t^i) p(\mathbf{s}_t^i|\mathbf{s}_{t-1}^i)}{q(\mathbf{s}_t^i|\mathbf{s}_{t-1}^i, \mathbf{z}_t)}. \tag{3.3.4}$$

It can be shown that, as the number of particles N approaches ∞, the approximation in equation 3.3.3 approaches the true posterior PDF $p(\mathbf{s}_t|\mathbf{z}_{1:t})$. The SIS particle filtering algorithm is now given by Algorithm 1.

The common problem with the SIS particle filter is that after a few iterations, all but one particle will have negligible weight. This is known as the degeneracy phenomenon which researchers have shown that it is impossible to avoid [29]. This means that a large portion of computational time will be spent on updating particles with weight values that are close to zero. The sequential importance resampling algorithm is used to address this phenomenon.

Algorithm 2 The resampling algorithm [9].

$[\{\mathbf{s}_t^{j*}, w_t^j, i^j\}_{j=1}^N] = \texttt{RESAMPLE}[\{\mathbf{s}_t^i, w_t^i\}_{j=1}^N]$

- Initialize the CDF: $c_1 = 0$
- FOR $i = 2 : N$
  - Construct CDF: $c_i = c_{i-1} + w_t^i$
- END FOR
- Start at the bottom of the CDF: $i = 1$
- Draw a starting Point: $u_1 \sim \texttt{U}[0, N^{-1}]$
- FOR $j = 1 : N$
  - Move along the CDF: $u_j = u_1 + N^{-1}(j-1)$
  - WHILE $u_j > c_i$
    - i=i+1
  - END WHILE
  - Assign sample: $\mathbf{s}_t^{j*} = \mathbf{s}_t^i$
  - Assign weight: $w_t^j = N^{-1}$
  - Assign parent: $i^j = i$
- END FOR

3.3.4 Sequential Importance Resampling

Sequential importance resampling is a means to prune particles with low weight values and replace them with significant particles. These significant particles will most likely be duplicated multiple times from particles with large weights, and conversely, particles with very small weights are not likely to be duplicated at all. After resampling, the weights of all particles will all be equal to 1/N as shown in Figure 3.3.1. The resampling algorithm is given in Algorithm 2.

The sequential importance resampling algorithm has a few disadvantages. The first is that the ability to parallelize the algorithm is lost because particles are dependent on other particles so they cannot be put into separate processes that could run in parallel. This is due to the spawning of new particles from previous particles which make a significant impact on the total computational time. The other drawback of the algorithm is that particle diversity decreases after each resampling stage. This means that, statistically, more particles will spawn from other particles with the highest weight values, resulting in a focus on those particles and limiting the search space around them.

3.3.5 The General Particle Filter Algorithm

For location tracking problems where an application seeks to estimate location information, $p(\mathbf{s}_{0:t}|\mathbf{z}_{1:t})$ is not available to be used in a SMC simulation. Thus, one must make use of an importance PDF $q(\cdot)$ that is related to $p(\mathbf{s}_{0:t}|\mathbf{z}_{1:t})$. In the context of RSSI-based localization, the training dataset can be used to generate the importance PDF. The general algorithm for the Particle filter is given in Algorithm 3.

3.4 Conclusion

In this chapter, three fingerprinting algorithms were explained that can be used for RSSI-based fingerprinting localization. Each has advantages and disadvantages with respect to complexity and performance. In the next chapter, each algorithm is tested on multichannel real RSSI data collected in an indoor environment. Ultimately, the goal of our effort is to improve localization performance with multichannel RSSI.

Algorithm 3 Particle Filter

$[\{\mathbf{s}_t^i, \omega_t^i\}_{i=1}^N] = \texttt{PF}[\{\mathbf{s}_{t-1}^i, \omega_{t-1}^i\}_{i=1}^N, \mathbf{z}_t]$

- FOR $i = 1 : N$
  - Draw $\mathbf{s}_t^i \sim q(\mathbf{s}_t^i | \mathbf{s}_{t-1}^i, \mathbf{z}_t)$
  - Assign the particle a weight, ω_t^i, according to equation 3.3.4
- END FOR
- Calculate the total weight: $total = \texttt{SUM}[\{\omega_t^i\}_{i=1}^N]$
- FOR $i = 1 : N$
  - Normalize: $\omega_t^i = (total)^{-1} \omega_t^i$
- END FOR
- Resample using Algorithm 2

Chapter 4

Results

This chapter explores three fingerprinting methods for RSSI-based 2-dimensional localization and then compares the performance of each method while discussing their strengths and weaknesses. Each of the fingerprinting methods perform localization using frequency diversity by collecting RSSI measurements over multiple communication channels in an attempt to mitigate localization error.

The chapter begins by describing the data collection process and provides details on the experimental set-up. A performance comparison of k-nearest neighbor (kNN), artificial neural network (ANN), and particle filter (PF) is then presented near the end of the chapter.

4.1 Experimental Setup and Data Collection

The user must perform calibration before using any of the fingerprinting techniques considered in this chapter. During calibration, a training dataset containing RSSI measurements with corresponding known recorded locations is constructed. Only a single transmitter and receiver are used during the data collection stage in order to demonstrate the performance of multichannel fingerprinting based localization. That is, only one static *anchor* and one mobile *target* are used. These results can easily extend to larger systems with multiple anchors and targets. Additionally, the entire data collection process was performed in the smart-space of the Perceptual Systems Research Group (PSRG) lab.

A TI ez430-Chronos smart watch served as the mobile target for the data collection process and was chosen due to its convenient form factor, as shown in Figure 4.1.1. The watch uses a CC430F6137 TI microcontroller which operates at the 915 MHz ISM band. The watch requires a 3 volt CR2032 battery to function and comprises many on-board sensors including a barometer, accelerometer, and a thermometer. For the localization experiment, the watch only serves the purpose of a mobile, radio-enabled target.

The anchor used for the data collection is an Angelos Ambient: a custom radio enabled device shown in Figure 4.1.1 that was created by PSRG researchers. The anchor has a USB connector so that a standard 5 V power supply can be used to power the device. This allows the anchor to be installed anywhere that a standard 120 V electrical wall outlet is available by using an inexpensive AC to DC converter. The anchor also uses a CC1101 radio with firmware designed to continuously listen for data packets transmitted from the watch and forward them, along with the RSSI, to a nearby base station. The base station then uploads the RSSI information to a central database, making RSSI available to localization algorithms.

The entire data collection process was performed in the PSRG lab smart-space. The room's dimensions are 4.34 meters wide, 9.58 meters long, and 2.84 meters in height. The room also has a bed, reclining chair, plasma TV, desk, and other items as shown in Figure 4.1.2. This room is used to perform health monitoring experiments and is designed to resemble a typical person's bedroom.

Figure 4.1.1: The TI's ez430-Chronos smart watch and Angelos Ambient

Figure 4.1.2: The smart space was used to conduct the experiment.

The anchor was placed on one side of the smart-space and the watch was placed near the center during the data collection process. The anchor was mounted on a wooden rod to keep it at a height of one meter. The watch was mounted on a wooden fixture placed on top of a movable cart and was also kept at a height of one meter. It is important for both devices to be kept at a height of one meter throughout the experiment to eliminate variability associated with vertical movement. A diagram of the top view of the smart-space can be seen in Figure 4.1.3. The blue circle represents the position of the anchor while the green square represents the 1 meter by 1 meter

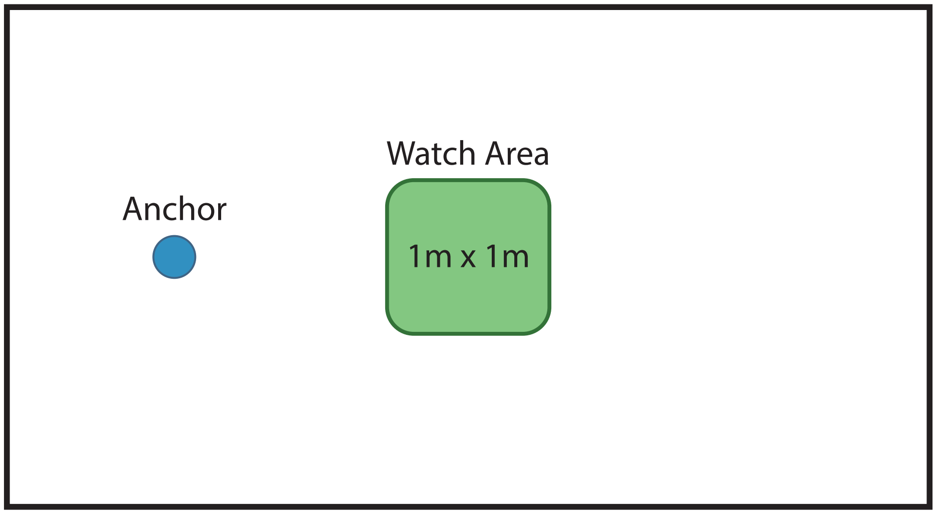


Figure 4.1.3: Basic room layout for RSSI data collection

area that the watch occupied during the data collection stage.

It should be noted that the 1 meter by 1 meter area occupied by the watch is relatively small when compared to the entire room, but there were a few reasons why the area was kept to this size. First, it was necessary to capture transitions between local extrema in RSSI caused from multipath fading. Figure 4.1.4, shows a plot of the mean and standard deviation of RSSI versus distance for line-of-sight between the watch and anchor in the smart-space for three different channels. The mean and standard deviation were computed for 100 RSSI measurements at each location and for each channel. The variance is denoted by the the light shaded area around the data lines. The blue line displays data for channel 96, the red for 116, and the green for 136. All three lines show that the measured RSSI separately follows the path-loss model in a multipath environment as described in Chapter 1 with all lines appearing noisy and displaying a high number of local extrema for short distances due to fast fading. To capture the transition between local minima and maxima, one must sample the area with a high spatial resolution. In order to determine an acceptable sampling

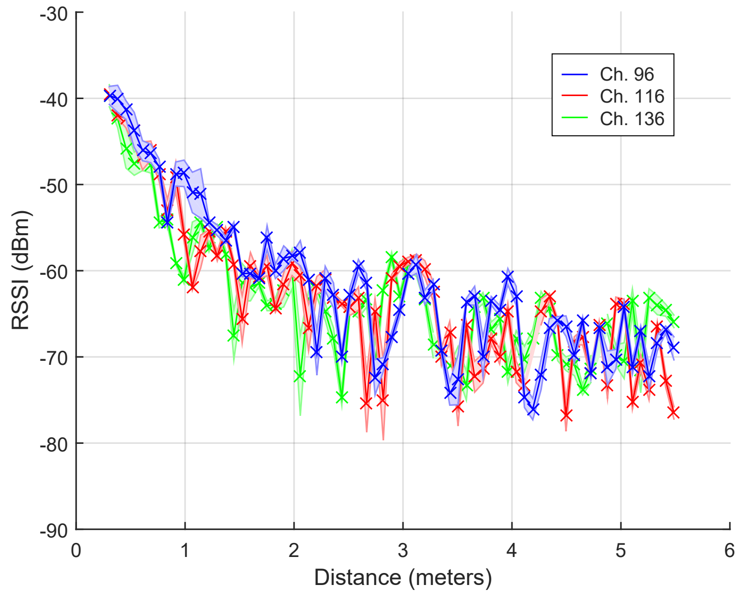


Figure 4.1.4: RSSI versus distance for line-of-sight between the anchor and watch

resolution, the RSSI plots as shown in Figure 4.1.5 were created with 2.5 centimeter, 7.5 centimeter, 12.5 centimeter, and 17.75 centimeter resolutions. One may determine the minimum spatial resolution necessary by observing the differences between the four plots such as the increase and decrease of extrema. In fact, the occurrence of extrema reduces as the resolution decreases. At resolution lower than 7.5 centimeter (about 3 inches), the data already start to lose many of the minima and maxima present in the 2.5 centimeter plot. This observation keeps occurring as the resolution continues to decrease. The sampling resolution was judged to be acceptable if it captured most transitions between the extrema. This requires a resolution less than 7.5 centimeter and we chose a resolution 2 centimeter to ensure that these conditions were met.

Due to the required high spatial resolution and a sampling rate limited to 6 Hz, the data collection process required 36 minutes to cover the 1 meter by 1 meter area.

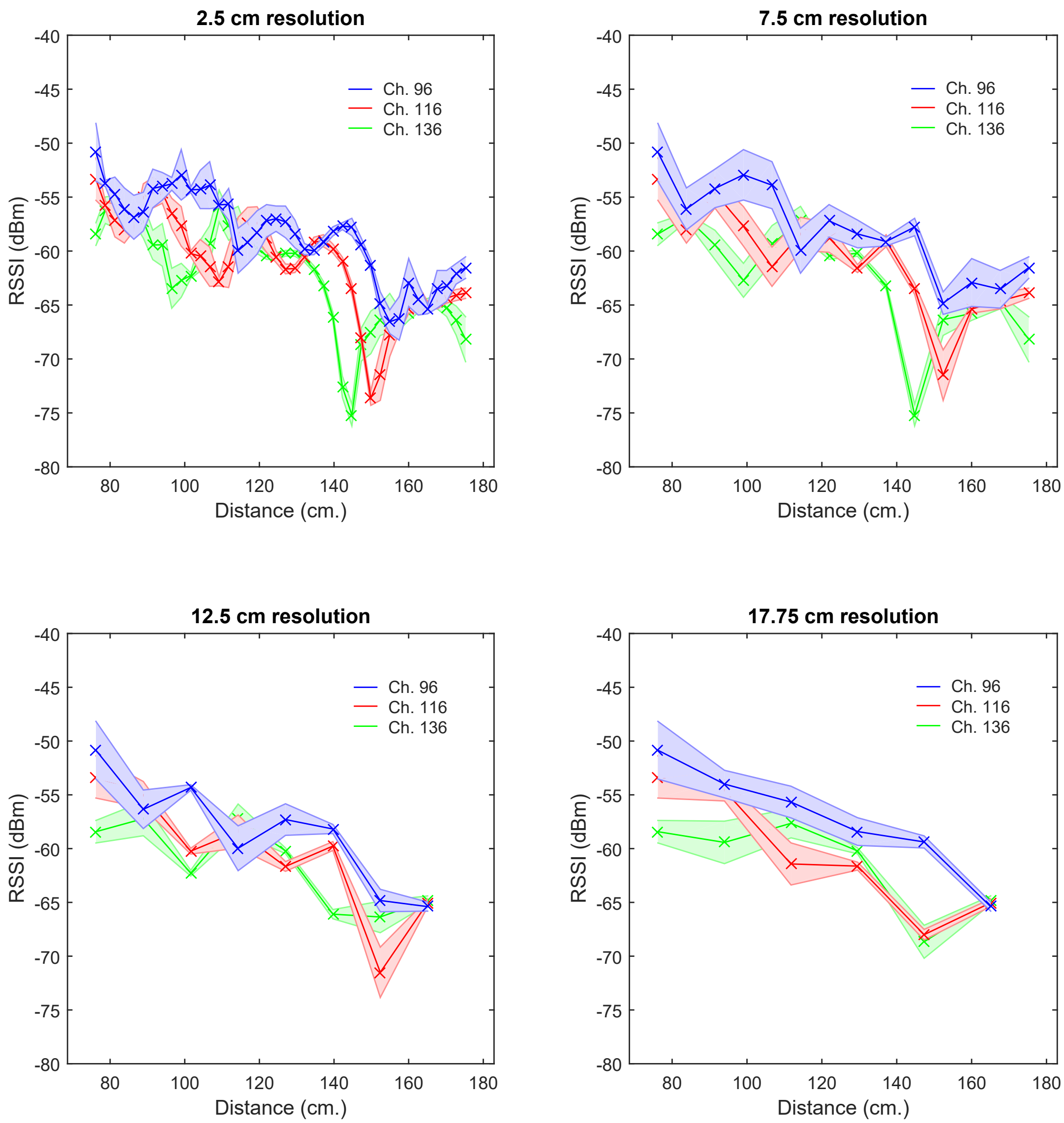


Figure 4.1.5: RSSI versus distance for different sampling resolutions

It would have taken about an hour and a half to sample a 1.5 meter by 1.5 meter area and if a 2 meter by 2 meter area is sampled, that would require 2 hours. The total area covered by the watch was kept at 1 meter by 1 meter to limit the amount of time spent during the data collection process.

Figures 4.1.4 and 4.1.5 show that the RSSI at each location is relatively consistent since the standard deviation is low at each location. This means that the RSSI is not extremely noisy within the environment. The light color shaded area around the data lines in Figures 4.1.4 and 4.1.5 indicate the standard deviation.

4.1.1 Channel Selection

Data packets were transmitted by the watch to the anchor over ten communication channels with a protocol that uses Gaussian frequency shift keying (GFSK) modulation. The highest channel was centered at 835.7 MHz and the lowest channel was centered at 918.14 MHz. The channel spacing was 374 kHz and the frequency deviation was 128 kHz. Channels were chosen to be relatively evenly spaced within the available frequency band while avoiding the following channels:

- 65-90 (856.3 - 865.6 MHz)
- 145-150 (886.3 - 888.1 MHz)
- 160-168 (891.9 - 894.9 MHz)

due to possible communication interference. When scanning the 918 MHz ISM with a 8591E Hewlett-Packard spectrum analyzer, it was found that activity of RF transmission was present within the three above frequency ranges by unknown devices. There were two reasons to avoid using the channels: 1) to not interfere with other

Channel Order	Actual Channel	Center Frequency (MHz)
1	230	918.14
2	50	850.73
3	136	882.93
4	10	835.75
5	191	903.53
6	96	867.95
7	171	896.04
8	30	843.24
9	116	875.44
10	211	911.02

Table 4.1: List of channels used during the data collection stage

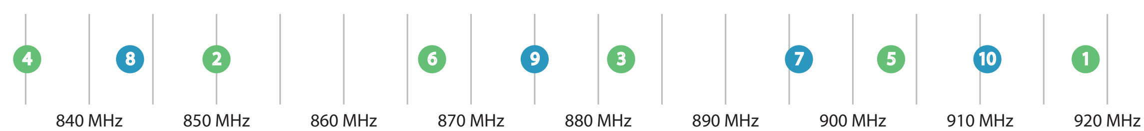


Figure 4.1.6: Visual representation of the channels listed in Table 4.1

radio devices, and 2) to avoid possible packet loss due to transmitting data in a high interference channel.

The ten selected channels are listed in Table 4.1 which also lists each channel's numerical order, actual channel number, and center frequency. Channel 10 is the channel with lowest center frequency (835.7 MHz) and channel 230 has the highest center frequency (918.14 MHz).

Figure 4.1.6 shows a visual representation of the channels listed in Table 4.1 with in the allowed frequency band. The numbered circles are the channels. The number in the center of each circle is the channel label from the table. The first six channels that are listed in Table 4.1 are indicated by green circles. The other four channels are indicated by blue circles.

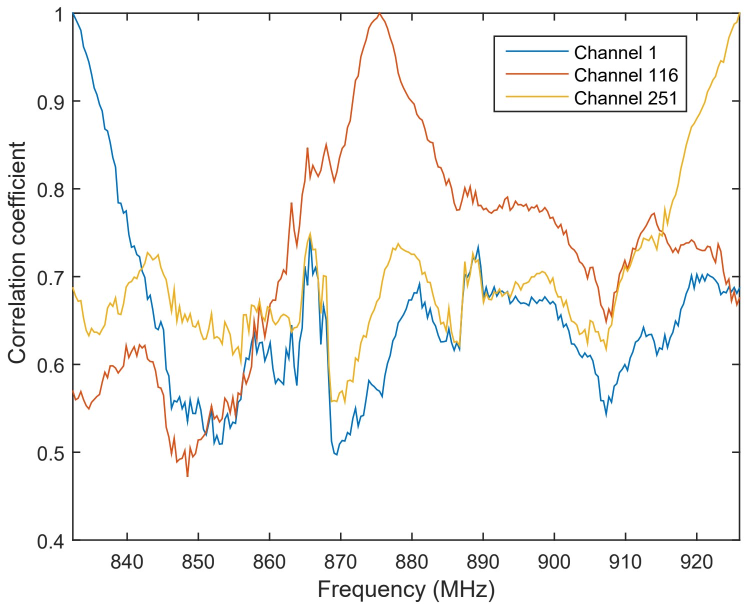


Figure 4.1.7: Pearson correlation coefficient between channels 1, 116, and 251 with all other channels

The organization for the channel order was inspired by a simple experiment that was performed to explore the idea of channel RSSI cross-correlation. The data collection process in Figure 4.1.4 was repeated for RSSI on all communication channels between 1 and 251. Only three of those channels were chosen to compute the Pearson correlation coefficient with all other channels using the RSSI versus distance data lines. Figure 4.1.7 shows the results for channels 1, 116, and 251.

As shown in Figure 4.1.7, the correlation coefficient equals one for the case where the coefficient is computed for a channel with itself. The figure also shows a high correlation amongst neighboring channels that drops as you move further away in any direction within the frequency band. These results strongly suggest that choosing channels that are distant from one another correlate less and thus can provide the most information when considered jointly. This knowledge was taken into account when organizing the channels.

4.1.2 The Dataset used for the Localization Algorithms

The watch was moved around the 1 meter by 1 meter area shown in Figure 4.1.3 while continuously transmitting data packets. The orientation of the watch was kept the same throughout the entire data collection stage to remove variations caused by antenna directionality. At the top of the wooden fixture holding the watch, a UTM-30LX Hokuyo laser range finder was mounted to provide highly accurate watch location measurements that were used as the ground-truth for each of the RSSI measurements. The Hokuyo laser range finder has an accuracy of 3 centimeter or 1.18 inches.

While the watch transmitted data packets on all ten channels, six times a second, the cart was moved around the entire test area. Figures 4.1.8 and 4.1.9 show the RSSI collected within the 1 meter by 1 meter area for all ten channels. Each data point is represented by a colored circle depicting RSSI measured by the anchor for that location. Additionally, each figure has a color-bar that indicates RSSI intensity. These measurements were suitable to be used as the training dataset because they had sufficient spatial resolution to capture the majority of extrema that occur within the 1 meter by 1 meter area.

As described in Chapter 3, a separate testing dataset was created to evaluate the performance of each algorithm and to ensure that the training and testing sets are mutually exclusive. The testing dataset consists of data points of a shorter path within the 1 meter by 1 meter area as shown in Figure 4.1.10. The training dataset was composed of 7,084 RSSI measurements while the testing dataset consisted of 1288 measurements. The color bar in Figure 4.1.10 indicates time progressing. The dark

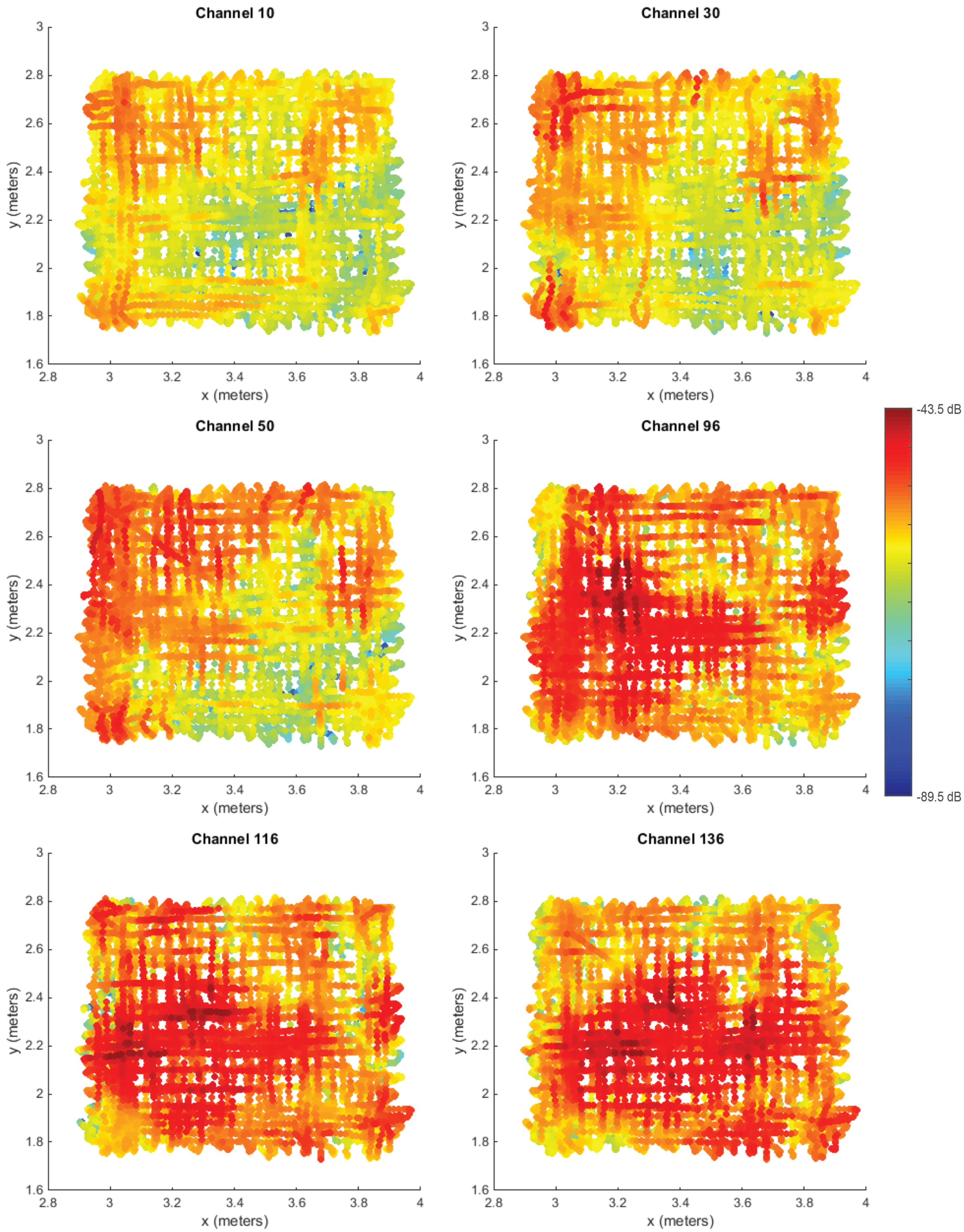


Figure 4.1.8: RSSI heat-maps for the first six channels

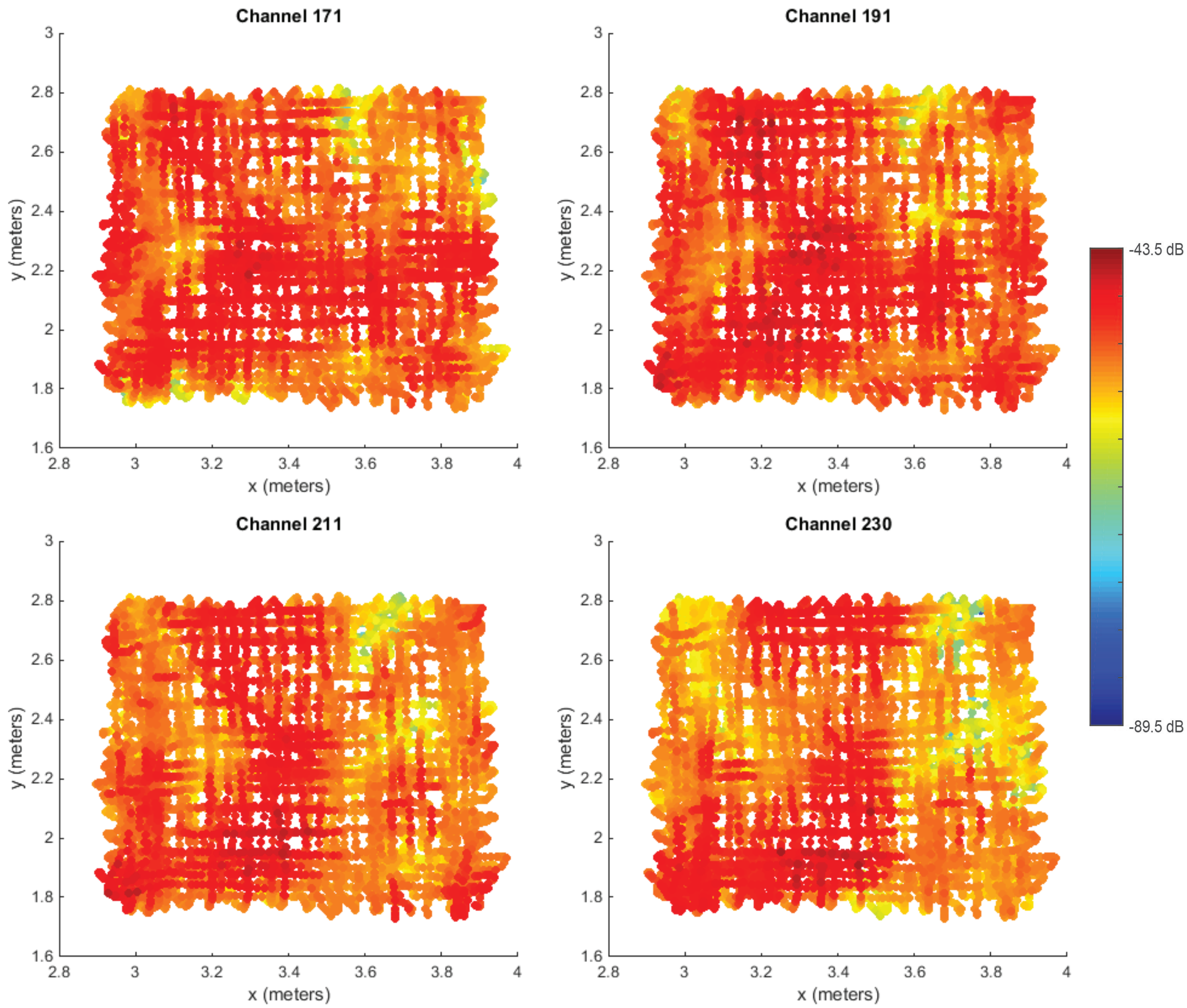


Figure 4.1.9: RSSI heat-maps for the last four channels

blue color corresponds to the beginning of the testing dataset and the light yellow color corresponds to the end of the dataset.

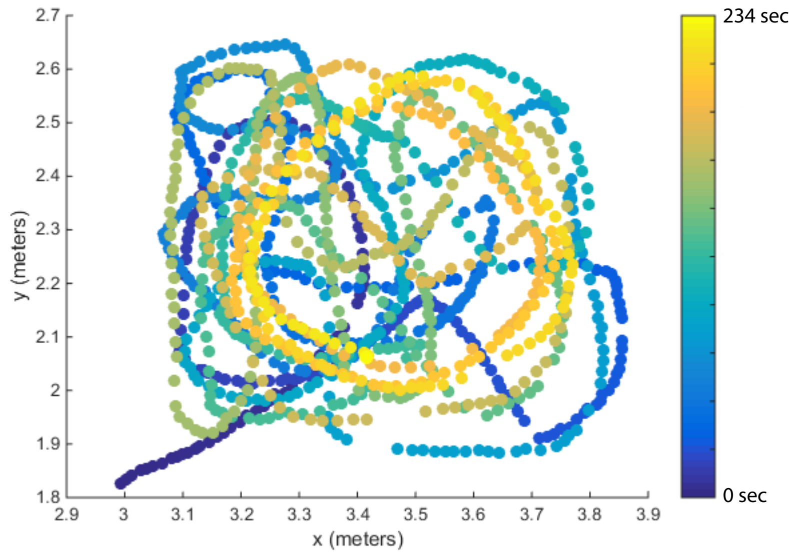


Figure 4.1.10: Testing dataset

4.2 Training and Testing Datasets with Localization Algorithms

The three algorithms are trained using the training dataset and their performance was evaluated using the testing dataset. Each algorithm was then evaluated based on the accuracy of its prediction by computing the average Euclidean distance between the estimated and true location. The ultimate goal is to determine whether or not multichannel data helps reduce prediction error. However, prior to the analysis, an explanation of design decisions and parameter selection are given for all algorithms.

4.2.1 k-Nearest Neighbor Algorithm for RSSI-Based Localization

The number of nearest neighbors indicates the total number of closest data points that are used to compute a location estimate through weighted averaging. As explained

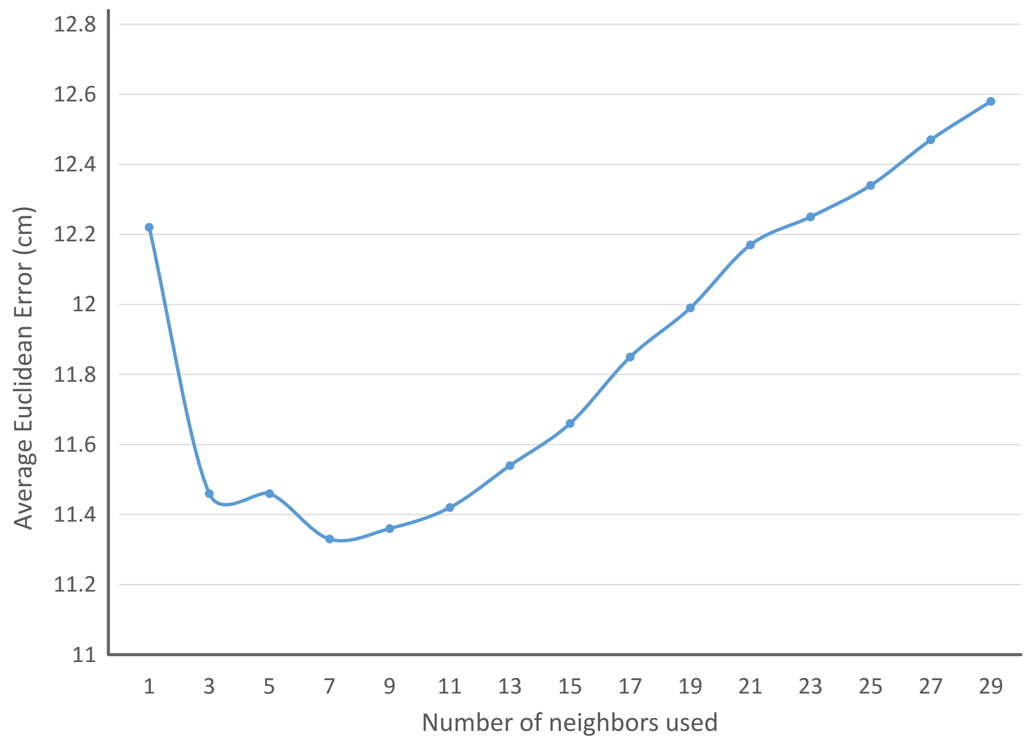


Figure 4.2.1: kNN performance as a function of the neighbors used

in Chapter 3, this is the only parameter of the kNN algorithm and when given a training and testing dataset, the algorithms finds the k closest data points in the training dataset for each data point in the testing dataset. The coordinates of the k closest points are then averaged, giving a location estimate of the watch. The number of nearest neighbors was varied for the algorithm over a large range of values as shown in Figure 4.2.1 where the kNN's performance is the y-axis and the number of neighbors is the x-axis. The performance is given by the average Euclidean error

$$\bar{E}(\mathbf{e}) = mean(|\mathbf{e}|),$$

where $\mathbf{e} = \mathbf{s}_{1:N} - \hat{\mathbf{s}}_{1:N}$ and $\bar{E}$ is average distance between an estimate and the true location of the watch.

The best performance is seen when the number of neighbors is within the range of 7 and 9, which results in an $\bar{E}$ of less than 11.4 centimeter. Outside of this range, the kNN's performance degrades and decreases. There are two different reasons why this happens, each corresponding to whether a the number of neighbors uses is higher or lower than 7 or higher than 9. When using a low value, such as 1, one will not have enough data points to mitigate error due to noise through location averaging. When using a high value, such as greater than 11, then the algorithm starts using too many data points where the furthest points start skewing the results. A nice balance between the two effects is seen within a certain range. Since the best performance was observed when using 7 neighbors ($\bar{E}$ of 13.25 centimeter), this was chosen as the the default parameter value for all subsequent computations.

RSSI data from all ten channels was used for this tuning process. This ensured that the algorithm was tuned with all available information. The next section demonstrates that doing this does not negatively impact the algorithm's performance, and in fact, improves the $\bar{E}$.

4.2.2 Artificial Neural Network for RSSI-Based Localization

The three-layer artificial neural network (ANN) only has one variable parameter: the number of neurons in network's hidden layer, or the hidden layer size. The neurons are responsible for learning a number of features from the training dataset that contribute to the final output. The algorithm was used while varying the hidden layer size as shown in Figure 4.2.2 to determine a suitable parameter value. The number of neurons in the hidden layer was varied between one and 300. The x-axis in Figure 4.2.2 indicates the hidden layer size. The blue, orange, and gray lines indicate the ANN's performance on the training, validation, and testing datasets respectively.

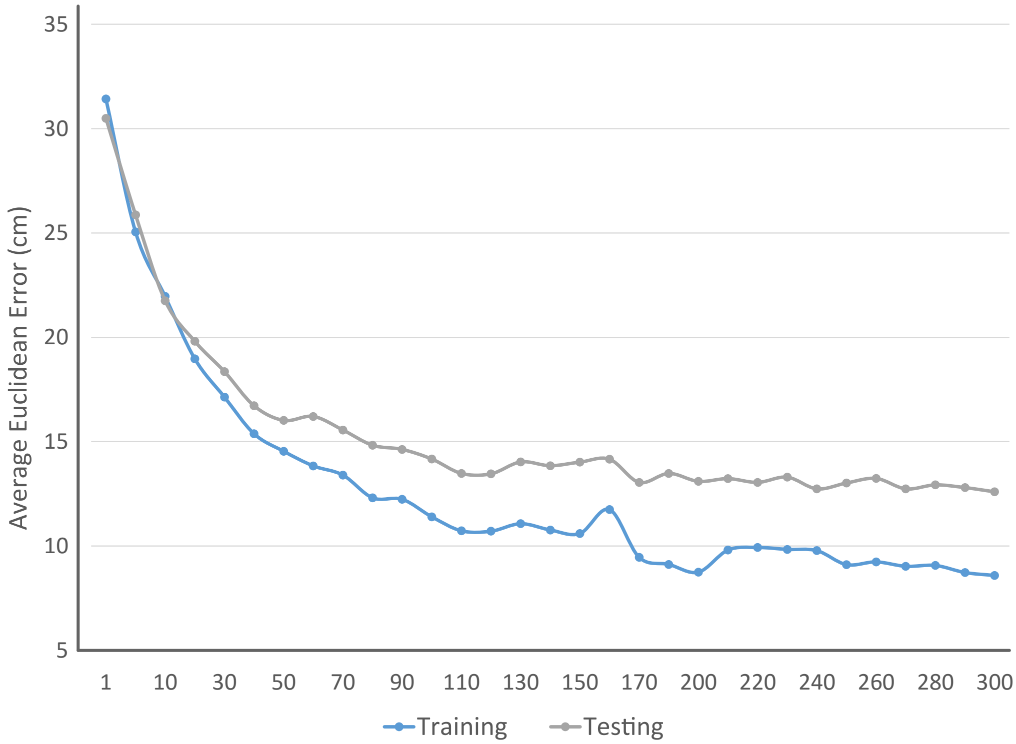


Figure 4.2.2: ANN performance as a function of the hidden layer size

Once again, RSSI from all ten channels were used during parameter tuning.

As the size of the ANN's hidden layer increases, so does its performance. An 18 centimeter average Euclidean error improvement is seen when using 200 neurons instead of one; a little over half of the original error. With more neurons, more features are learned from the training dataset which play a large role in disambiguating similar RSSI data measurements and improves localization performance. A suitable hidden layer size parameter value is chosen to be 70 for all subsequent computations. Choosing a value larger than 70 does not significantly increase the algorithm's $\bar{E}$, but increases the training time by about 5 minutes for every 70 neurons.

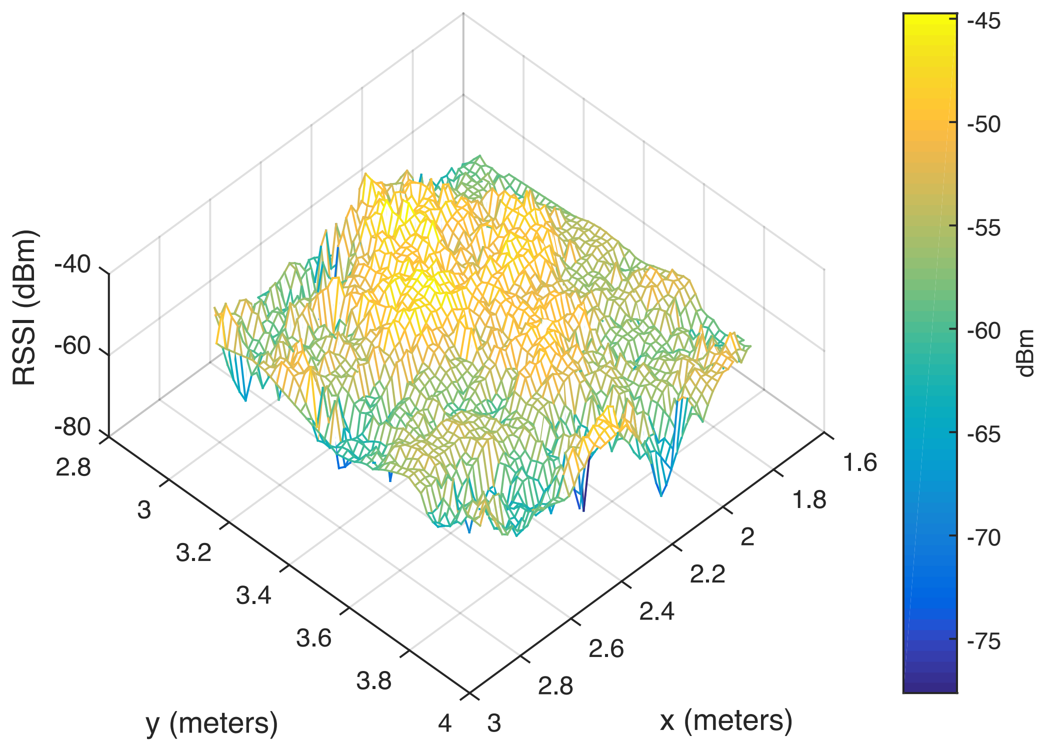


Figure 4.2.3: Three dimensional interpolated heat map

4.2.3 Particle Filter for RSSI-Based Localization

Finally, the PF was used to provide location estimates with the RSSI training and testing datasets. Here, the training dataset was first used to create RSSI maps that provided measurements for each filter particle. These measurements are artificial RSSI samples interpolated from these multichannel RSSI maps. It was important to use this interpolation method because there were a finite number of samples in the training dataset and the randomly moving particles could take on an infinite number of locations. MATLAB's interpolate scattered data function accomplished this for the 1 meter by 1 meter area as shown in Figure 4.2.3. A gridded map was created for each of of the channels in the training dataset.

4.2.3.1 Parameter Tuning for the Particle Filter

The PF has two parameters that can be varied: the number of particles and the measurement standard deviation. The number of particles used gives the PF the ability to search the RSSI maps and find the most probable locations for each new measurement $\mathbf{z}_t$. The number of particles used was varied from 25 to 200 particles. Figure 4.2.4 shows the performance of the $\bar{E}$ as a function of the number of particles. Immediately, one notices that the performance increases as the number of particles increase. With a higher number of particles "searching" the area, there are more measurements from which to compare the observation making it more likely to find the true location. Here, a 10 centimeter decrease in the $\bar{E}$ occurs when using 200 particles instead of 25. Even though a high number of particles corresponds to a higher location accuracy, there exists a trade-off between decreasing the $\bar{E}$ and increasing the computation time. A suitable value for the number of particles was chosen to be 100 because using a higher values does not significantly increase accuracy.

The other variable parameter is the measurement SD which assigns the width of the multivariate Gaussian function that is used to compute the weight for each particle. The multivariate Gaussian function determines the similarity between the particles and $\mathbf{z}_t$ during the resampling step. The multivariate function was chosen due to its simplicity and although it may not be an optimal choice due to RSSI's non-Gaussian nature, these results show that the function works well. The measurement SD was varied between the range of 0.01 and 3000 as shown in Figure 4.2.5.

The PF algorithm performs best when the measurement SD lies between 0.3 and 100. Outside of this range, the PF's performance degrades by at least 10 centimeter. When using low measurement SD values, only the particles that are close to $\mathbf{z}_t$ will receive large weight value. This starts to limit the spatial diversity that the particles

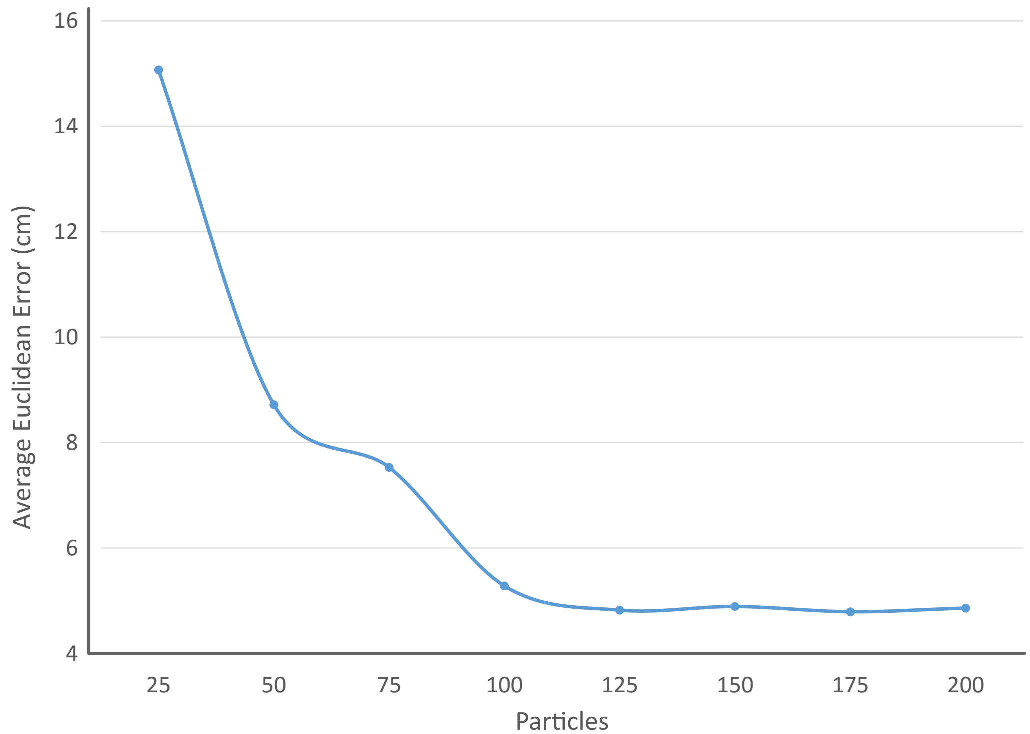


Figure 4.2.4: PF performance as a function of the number of particles used

can take on, where their space becomes limited after each resampling step. And, since RSSI is noisy, it affects the PF's performance during averaging. On the other hand, using a high measurement SD value will make the PF take a larger number of particles into consideration when averaging. The additional particles may have locations that are farther away from the true location of the watch. Because a measurement SD of 30 rendered the lowest $\bar{E}$, this value was chosen for all subsequent experiments.

4.2.3.2 The Particle Filter Delay

An interesting observation is that the PF's performance improves if the location estimate is delayed. In other words, the $\bar{E}$ decreased when a current estimate was compared to a past true location of an observation. To show this, the PF's performance is

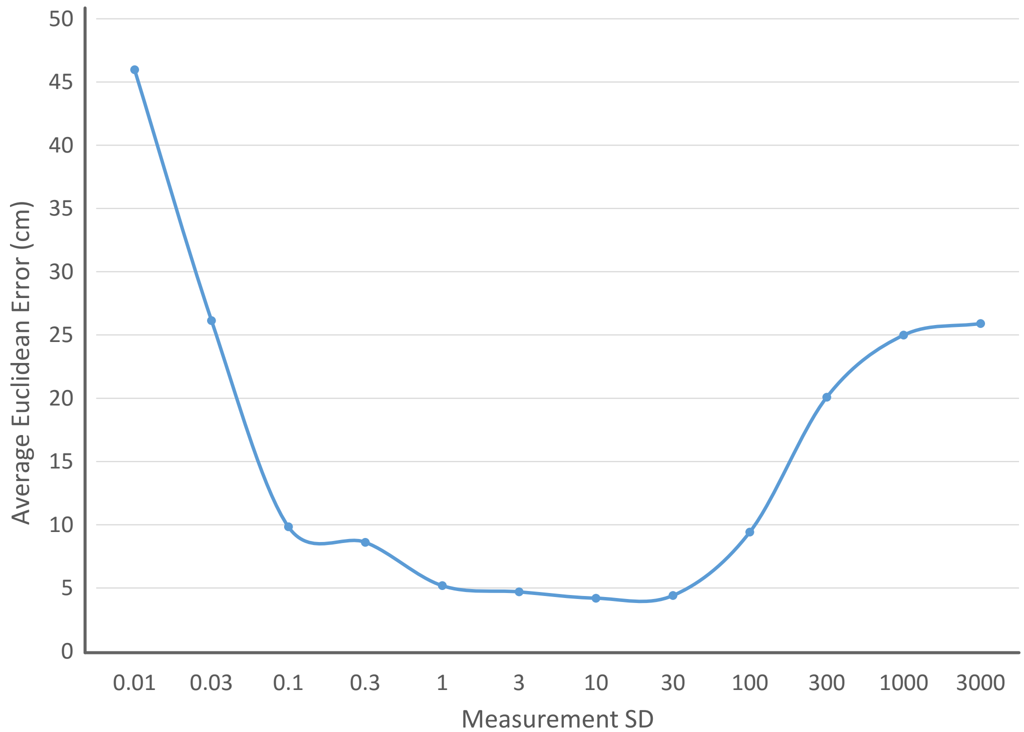


Figure 4.2.5: PF performance as a function of the measurement SD

shown with various time-delays in Figure 4.2.6. The real time estimate lagged behind the true location most of the time during the experiment. This phenomenon occurs due to making no assumptions about the state transition model other than random movement. Most PF tracking applications assume some movement model that adds useful information, such as designing a model around the knowledge of the acceleration or velocity for the moving watch. In the future, data from the accelerometer could prove useful in improving the PF performance and make it a true real-time system. It was observed that an offset of five rendered the lowest $\bar{E}$. The measurement SD was kept at this value for all subsequent computations.

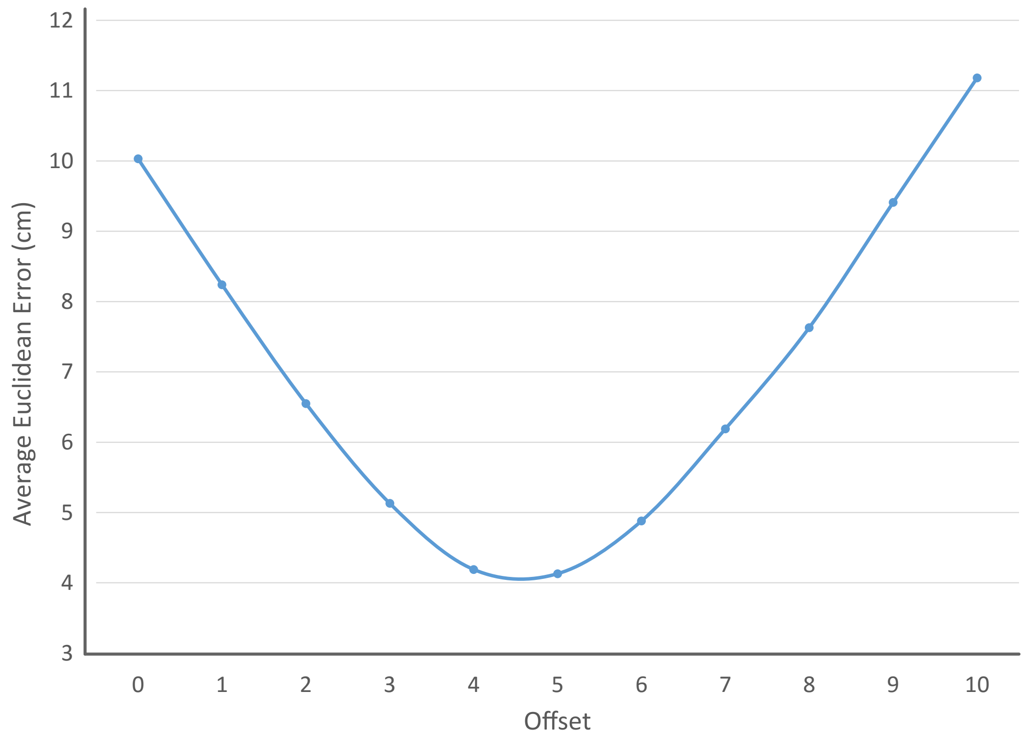


Figure 4.2.6: PF performance as a function of an output offset

4.3 Comparison of the Three Algorithms

Figure 4.3.1 shows the performance for all three algorithms as a function of the number of channels used in the training and testing datasets. The blue line corresponds to the kNN algorithm's performance, the orange corresponds to the ANN, and the gray to the PF. The x-axis indicates the number of channels used. All of the algorithms are evaluated by computing the $\bar{E}$ of their location estimates.

The PF performs the best with at least 5 centimeter $\bar{E}$ improvement over the other two algorithms using any number of channels in the training and testing dataset. The ANN has the worst performance in all cases except when using a single channel, where the kNN's $\bar{E}$ is also the highest. The kNN is the second best algorithm when using two or more channels.

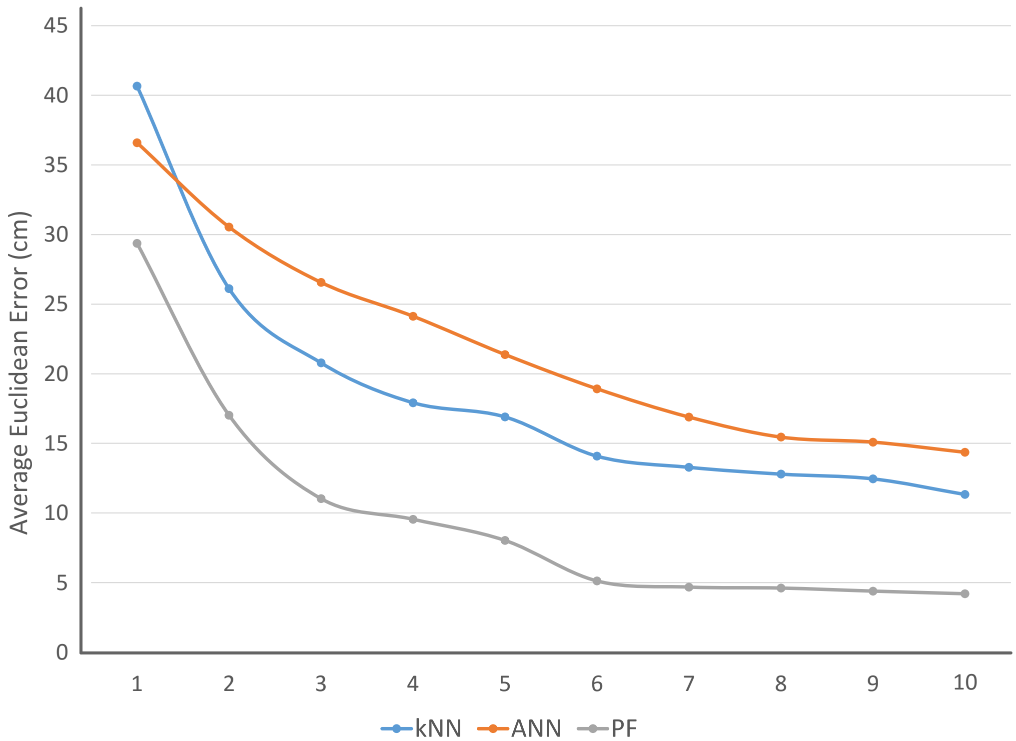


Figure 4.3.1: kNN, ANN, and PF performance as a function of channels

The PF offers the best performance because it takes advantage of the temporal aspect during the location estimation. It makes a location estimate that is influenced by the previous location estimate. The particles are first initialized randomly, but after a number of iterations, they congregate to a general area as a result of the resampling step. Knowledge of the particle's previous position is used during the particle update step. The PF is a time-dependent system whereas the other two algorithms are time-invariant.

The PF can give a location estimate with an average Euclidean error of 4.5 centimeter. However, it must be considered that the algorithm performs well under the constraints imposed during the experimental design. For instance, a person was not wearing the watch. Instead, the watch was mounted on a wooden fixture placed on top of a movable cart with the same orientation throughout the whole experiment. Even though the watch has an omnidirectional antenna radiation pattern, a person wearing the watch will alter the RSSI by attenuating RF radiation with their body placed between the anchor and watch [81]. Also, the sampled area was only 1 meter by 1 meter, which is not a large area but which was kept at that size because it was desired to capture transitions between local extrema with an acceptable sampling rate. Theoretically, the sampled area can be increased by using a lower RF frequency band during communication. This will increase the average distance between local extrema but also scales up the average Euclidean error in the algorithm's location estimate.

Regardless of the method used, it is impressive that 2-dimensional localization is performed with RSSI measurements from a single anchor and watch; a great advantage of fingerprinting techniques. A log-normal based model, used in many traditional methods, simply can not do this because they only infer a distance between a target and anchor from RSSI. The target could then lie anywhere on a circle centered around the anchor with a radius equal to the estimated distance. The results of this work even suggest that fingerprinting can even achieve 3-dimensional localization if the calibration stage is designed to include 3-dimensional data. This system can also easily be scaled up to include more watches and anchors. This adds the ability to track multiple targets and most likely increase localization accuracy through receiver diversity. A disadvantage of using multiple communication channels is that it significantly reduces the watch's battery life. For example, power consumption is doubled by using two communication channels rather than just one.

Table 4.2 lists other indoor localization systems in order of their localization accuracy. The table shows that most of the systems surpass the performance of our multichannel fingerprinting approach; some of these systems can even perform 3-

System Name	Technology	Accuracy	Localization Type
UTM-30LX	Laser	1 cm	2-dimensional
MIT's Cricket	Ultrasonic	1-3 cm	3-dimensional
Cambridge's Bat	Ultrasonic	3 cm	3-dimensional
DW1000	UWB TDOA	10 cm	3-dimensional
Ubisense	UWB TDOA	15 cm	3-dimensional
Multichannel Fingerprinting	RSSI	4.5-15 cm	2-dimensional

Table 4.2: Accuracy of indoor localization systems.

dimentional localization. Even though these systems may seem superior to multichannel fingerprinting in terms of localization accuracy, they have higher cost and increased complexity compared to multichannel fingerprinting. For example, Hokuyo's UTM-30LX laser scanner [47] provides the finest resolution, but this laser scanner costs $5,000! MIT's Cricket [79] and Cambridge's Bat [21], which are ultrasonic based localization methods, have their disadvantage in that line-of-sight must always be established between the target and the anchor. DecaWave's DW1000 [77] and the Ubisense systems [119], which are ultra wideband (UWB) time-difference-of-arrival (TDOA) based systems, provide more accurate localization than multichannel fingerprinting, but they are relatively expensive due to the strict constraints on timing synchronization. In the end, multichannel fingerprinting provides a low cost, low power, and low complexity localization solution with reasonable accuracy.

Chapter 5

Conclusion

This work shows that using multiple channels can improve 2-dimensional localization accuracy for RF-based fingerprinting methods while still using low power and low cost hardware. The first chapter introduced indoor localization and provided the motivation for RSSI-based systems. The second chapter examined multipath propagation and presented current indoor RSSI-based localization solutions. Chapter 3 presented three different multichannel fingerprinting based algorithms that were developed to improve RSSI localization. Finally, Chapter 4 presented results that showed that the three methods successfully reduce the localization error compared to single channel systems.

Experiments showed that the best method, a particle filter, achieves a 4.5 centimeter average Euclidean error with 10 different communication channels on the 918 MHz ISM band. The second best method is kNN followed by the ANN. The particle filter performed better for multiple reasons. First, it is a time-dependent system that exploits knowledge of previous observations to make its current estimate. This is advantageous because adjacent measurements are dependent on each other's position with respect to time. Having this property leads to better performance in this case.

Second, the filter uses a Monte Carlo sampling method that approaches the optimum Bayesian solution. In other words, the PF approximates the optimal solution, while the kNN and ANN do not. Regardless of which method is used, the results demonstrate that 2-dimensional localization can be achieved using RSSI data from only a single target and anchor. This is not possible using traditional localization approaches that rely on direct measurements of distance.

This work demonstrates that fingerprinting techniques are promising, but the time required for the calibration is a significant disadvantage. This process could be automated through robotics to increase the coverage area. For example, a small rover robot equipped with a target and laser rangefinder can move around multiple rooms to collect data overnight to be used the next morning. Another drawback of fingerprinting methods is that they are sensitive to dynamic environments. Simple changes in the environment, such as moving furniture, will affect multipath propagation and change the RSSI maps for a room. Thus it is necessary to re-perform calibration every time objects are moved.

For future work, it would be interesting to investigate how performance is affected by scaling up the system to include more anchors. Introducing additional anchors will reduce the error in location estimates by providing additional anchor dependent RSSI maps that will have supplementary location information. It would also be interesting to perform 3-dimensional localization with the proposed system, which our results suggest is feasible. One only needs to vary the target's vertical position during calibration to sample a 3-dimensional area and create a 3-dimentional calibration dataset. In this case, the system would be performing 3-dimensional localization with a single target and anchor whereas traditional methods require at least four anchors for 3-dimensional localization. Additionally, it would be beneficial to investigate adding a time-dependent component to kNN and ANN. The PF was the most successful because it is a time-dependent system where the current prediction relied on previous information. Adding a time-dependent component to the ANN such as using a recurrent neural network structure should improve its location estimates. Finally, to expand the coverage area of the system, a lower frequency band can be used. This would result in RF signals with larger wavelengths that create more space between extrema transitions in RSSI maps. This would reduce the spatial sampling resolution, but would also introduce a trade off between using a lower frequency and having to use a larger antenna.

Bibliography

[1] Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mane, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viegas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *ArXiv e-prints*, 1603:arXiv:1603.04467, March 2016.

[2] B. Afsari, R. Tron, and R. Vidal. On the Convergence of Gradient Descent for Finding the Riemannian Center of Mass. *SIAM Journal on Control and Optimization*, 51(3):2230–2260, January 2013.

[3] Omar Ait Aider, Philippe Hoppenot, and Etienne Colle. A model-based method for indoor mobile robot localization using monocular vision and straight-line correspondences. *Robotics and Autonomous Systems*, 52(2-3):229–246, August 2005.

[4] Ian F. Akyildiz and Mehmet Can Vuran. *Wireless Sensor Networks*. John Wiley & Sons, Ltd, June 2010.

[5] S. Ali and P. Nobles. A Novel Indoor Location Sensing Mechanism for IEEE 802.11 b/g Wireless LAN. In *4th Workshop on Positioning, Navigation and Communication, 2007. WPNC '07*, pages 9–15, March 2007.

[6] M. Alwan, P.J. Rajendran, S. Kell, D. Mack, S. Dalal, M. Wolfe, and R. Felder. A Smart and Passive Floor-Vibration Based Fall Detector for Elderly. In *Information and Communication Technologies, 2006. ICTTA '06. 2nd*, volume 1, pages 1003–1007, 2006.

[7] J. Amores, N. Sebe, and P. Radeva. Boosting the distance estimation: Application to the K-Nearest Neighbor Classifier. *Pattern Recognition Letters*, 27(3):201–209, February 2006.

[8] K.R. Anne, K. Kyamakya, F. Erbas, C. Takenga, and J.C. Chedjou. GSM RSSI-based positioning using extended Kalman filter for training artificial neural networks. In *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, volume 6, pages 4141–4145 Vol. 6, September 2004.

[9] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, February 2002.

[10] A. Awad, T. Frunzke, and F. Dressler. Adaptive Distance Estimation and Localization in WSN using RSSI Measures. In *10th Euromicro Conference on Digital System Design Architectures, Methods and Tools, 2007. DSD 2007*, pages 471–478, August 2007.

[11] P. Bahl and V.N. Padmanabhan. RADAR: an in-building RF-based user location and tracking system. In *IEEE INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, volume 2, pages 775–784 vol.2, 2000.

[12] U. Bandara, M. Hasegawa, M. Inoue, Hiroyuki Morikawa, and T. Aoyama. Design and implementation of a Bluetooth signal strength based location sensing system. In *2004 IEEE Radio and Wireless Conference*, pages 319–322, September 2004.

[13] Andrea Bardella, Nicola Bui, Andrea Zanella, and Michele Zorzi. An Experimental Study on IEEE 802.15.4 Multichannel Transmission to Improve RSSI-Based Service Performance. In Pedro J. Marron, Thiemo Voigt, Peter Corke, and Luca Mottola, editors, *Real-World Wireless Sensor Networks*, number 6511 in Lecture Notes in Computer Science, pages 154–161. Springer Berlin Heidelberg, January 2010.

[14] I. A Basheer and M Hajmeer. Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods*, 43(1):3–31, December 2000.

[15] Roberto Battiti, Nhat Thang Le, and Alessandro Villani. Location-aware computing: a neural network model for determining location in wireless LANs. Departmental Technical Report, University of Trento, February 2002.

[16] B. Bejar, P. Belanovic, and S. Zazo. Distributed Gauss-Newton method for localization in Ad-hoc networks. In *2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, pages 1452–1454, November 2010.

[17] C.A. Boano, N. Tsiftes, T. Voigt, J. Brown, and U. Roedig. The Impact of Temperature on Outdoor Industrial Sensornet Applications. *IEEE Transactions on Industrial Informatics*, 6(3):451–459, August 2010.

[18] Martin Bor. *Indoor Localization for Wireless Sensor Networks: Applying multiple frequencies to mitigate multipath effects.* PhD thesis, TU Delft, Delft University of Technology, September 2009.

[19] Peter Brida, Peter Cepel, and Jan Duha. Geometric Algorithm for Received Signal Strength Based Mobile. *Radioengineering*, 14(2), June 2005.

[20] Jason Brownlee. Tutorial To Implement k-Nearest Neighbors in Python From Scratch, September 2014.

[21] AT&T Laboratories Cambrige. The Bat Ultrasonic Location System.

[22] J.D. Carlson, M. Mittek, S.A. Parkison, P. Sathler, D. Bayne, E.T. Psota, L.C. Perez, and S.J. Bonasera. Smart watch RSSI localization and refinement for behavioral classification using laser-SLAM for mapping and fingerprinting. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2173–2176, August 2014.

[23] E. Cassano, F. Florio, F. De Rango, and S. Marano. A performance comparison between ROC-RSSI and trilateration localization techniques for WPAN sensor networks in a real outdoor testbed. In *Wireless Telecommunications Symposium, 2009. WTS 2009*, pages 1–8, April 2009.

[24] Wenhaun Chi. A Revised Received Signal Strength Based Localization for Healthcare. *International Journal of Multimedia and Ubiquitous Engineering*, 10(10):273 – 282, 2015.

[25] Po-Jen Chuang and Yi-Jun Jiang. Effective neural network-based node localisation scheme for wireless sensor networks. *IET Wireless Sensor Systems*, 4(2):97–103, June 2014.

[26] Ronan Collobert, Koray Kavukcuoglu, and Clement Farabet. Torch7: A matlab-like environment for machine learning. 2011.

[27] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, January 1967.

[28] George Dedes and Andrew G Dempster. Indoor GPS Positioning Challenges and Opportunities. In *Proceedings of the IEEE Semiannual Vehicular Technology Conference*, September 2005.

[29] Arnaud Doucet. On sequential simulation-based methods for bayesian filtering. Technical report, 1998.

[30] Arnaud Doucet and Adam M. Johansen. A Tutorial on Particle Filtering and Smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12:656–704, December 2009.

[31] Robert Droual. Cells of the Nervous System, https://droualb.faculty.mjc.edu.

[32] Yeqing Fang, Zhongliang Deng, Chen Xue, Jichao Jiao, Hui Zeng, Ruoyu Zheng, and Shunbao Lu. Application of an Improved K Nearest Neighbor Algorithm in WiFi Indoor Positioning. In Jiadong Sun, Jingnan Liu, Shiwei Fan, and Xiaochun Lu, editors, *China Satellite Navigation Conference (CSNC) 2015 Proceedings: Volume III*, number 342 in Lecture Notes in Electrical Engineering, pages 517–524. Springer Berlin Heidelberg, 2015. DOI: 10.1007/978-3-662-46632-2_45.

[33] A. Fink and H. Beikirch. RSSI-based indoor localization using antenna diversity and plausibility filter. In *6th Workshop on Positioning, Navigation and Communication, 2009. WPNC 2009*, pages 159–165, March 2009.

[34] A. Fink, H. Beikirch, M. Voss, and C. Schroder. RSSI-based indoor positioning using diversity and Inertial Navigation. In *2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–7, September 2010.

[35] Peter A. Flach. *Machine Learning*. Number 10994. Springer, 2006.

[36] Henri P. Gavin. The Levenberg-Marquardt method for nonlinear least squares curve-fitting problems. Technical report, Department of Civil and Environmental Engineering, Duke University, September 2015.

[37] David E. Goldberg and John H. Holland. Genetic Algorithms and Machine Learning. *Machine Learning*, 3(2-3):95–99, October 1988.

[38] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *Radar and Signal Processing, IEE Proceedings F*, 140(2):107–113, April 1993.

[39] F. Gulmammadov. Analysis, modeling and compensation of bias drift in MEMS inertial sensors. In *4th International Conference on Recent Advances in Space Technologies, 2009. RAST '09*, pages 591–596, June 2009.

[40] Dongliang Guo, Yudong Zhang, Qiao Xiang, and Zhonghua Li. Improved Radio Frequency Identification Indoor Localization Method via Radial Basis Function Neural Network. *Mathematical Problems in Engineering*, 2014:e420482, July 2014.

[41] Zhang Guowei, Xu Zhan, and Liu Dan. Research and improvement on indoor localization based on RSSI fingerprint database and K-nearest neighbor points. In *2013 International Conference on Communications, Circuits and Systems (ICCCAS)*, volume 2, pages 68–71, November 2013.

[42] M.T. Hagan and M.B. Menhaj. Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks*, 5(6):989–993, November 1994.

[43] Simon S. Haykin and Michael Moher. *Modern Wireless Communications*. Pearson/Prentice Hall, 2005.

[44] M. Hazas and A. Hopper. Broadband ultrasonic location systems for improved indoor positioning. *IEEE Transactions on Mobile Computing*, 5(5):536–547, May 2006.

[45] K. Hechenbichler and K. Schliep. Weighted k-Nearest-Neighbor Techniques and Ordinal Classification. Working Paper, Ludwig Maximilians Universitat Munchen, 2004.

[46] Ghobad Heidari. *WiMedia UWB: Technology of Choice for Wireless USB and Bluetooth*. John Wiley & Sons, September 2008.

[47] LTD. Hokuyo Automatic CO. Hokuyo UTM-30lx Scanning Laser Rangefinder, https://acroname.com/products/.

[48] S. Holm, O.B. Hovind, S. Rostad, and R. Holm. Indoors data communications using airborne ultrasound. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05)*, volume 3, pages iii/957–iii/960 Vol. 3, March 2005.

[49] Nobuko Hongu, Jamie M. Wise, Barron J. Orr, and Kristin D. Wisneski. GPS Watches for Measuring Energy Expenditure during Physical Activity. *CALS Publications Archive. The University of Arizona.*, July 2014.

[50] Thomas Hornbeck, David Naylor, Alberto M. Segre, Geb Thomas, Ted Herman, and Philip M. Polgreen. Using Sensor Networks to Study the Effect of Peripatetic Healthcare Workers on the Spread of Hospital-Associated Infections. *Journal of Infectious Diseases*, page jis542, October 2012.

[51] Dong Hui-ying, Cao Bin, and Yang Yue-ping. Application of Particle Filter for Target Tracking in Wireless Sensor Networks. In *2010 International Conference on Communications and Mobile Computing (CMC)*, volume 3, pages 504–508, April 2010.

[52] Anil K. Jain, Jianchang Mao, and K. m Mohiuddin. Artificial Neural Networks: A Tutorial. *Computer*, 29(3):31–44, 1996.

[53] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. In *Proceedings of the 22Nd ACM International Conference on Multimedia*, MM '14, pages 675–678, New York, NY, USA, 2014. ACM.

[54] K. Kaemarungsi. Distribution of WLAN received signal strength indication for indoor location determination. In *2006 1st International Symposium on Wireless Pervasive Computing*, pages 6 pp.–, January 2006.

[55] Andrej Karpathy. CS231n Convolutional Neural Networks for Visual Recognition.

[56] K. Kasantikul, Chundi Xiu, Dongkai Yang, and Meng Yang. An enhanced technique for indoor navigation system based on WIFI-RSSI. In *2015 Seventh International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 513–518, July 2015.

[57] M. Khanafer, M. Guennoun, and H.T. Mouftah. WSN Architectures for Intelligent Transportation Systems. In *2009 3rd International Conference on New Technologies, Mobility and Security (NTMS)*, pages 1–8, December 2009.

[58] Seong Jin Kim and Byung Kook Kim. Dynamic Ultrasonic Hybrid Localization System for Indoor Mobile Robots. *IEEE Transactions on Industrial Electronics*, 60(10):4562–4573, October 2013.

[59] Diederik Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, December 2014. arXiv: 1412.6980.

[60] R. J. Kuo, M. C. Shieh, J. W. Zhang, and K. Y. Chen. The application of an artificial immune system-based back-propagation neural network with feature selection to an RFID positioning system. *Robotics and Computer-Integrated Manufacturing*, 29(6):431–438, December 2013.

[61] Wen-Hsing Kuo, Yun-Shen Chen, Gwei-Tai Jen, and Tai-Wei Lu. An intelligent positioning approach: RSSI-based indoor and outdoor localization scheme in Zigbee networks. In *2010 International Conference on Machine Learning and Cybernetics (ICMLC)*, volume 6, pages 2754–2759, July 2010.

[62] Jeril Kuriakose, Sandeep Joshi, R. Vikram Raju, and Aravind Kilaru. A Review on Localization in Wireless Sensor Networks. In Sabu M. Thampi, Alexander Gelbukh, and Jayanta Mukhopadhyay, editors, *Advances in Signal Processing*

and Intelligent Recognition Systems, number 264 in Advances in Intelligent Systems and Computing, pages 599–610. Springer International Publishing, 2014. DOI: 10.1007/978-3-319-04960-1_52.

[63] Branislav Kusy, Gyorgy Balogh, Janos Sallai, Akos Ledeczi, and Miklos Maroti. inTrack: High Precision Tracking of Mobile Sensor Nodes. In Koen Langendoen and Thiemo Voigt, editors, *Wireless Sensor Networks*, number 4373 in Lecture Notes in Computer Science, pages 51–66. Springer Berlin Heidelberg, 2007.

[64] Branislav Kusy, Akos Ledeczi, and Xenofon Koutsoukos. Tracking Mobile Nodes Using RF Doppler Shifts. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems*, SenSys '07, pages 29–42, New York, NY, USA, 2007. ACM.

[65] Branislav Kusy, Janos Sallai, Gyorgy Balogh, Akos Ledeczi, Vladimir Protopopescu, Johnny Tolliver, Frank DeNap, and Morey Parang. Radio Interferometric Tracking of Mobile Wireless Nodes. In *Proceedings of the 5th International Conference on Mobile Systems, Applications and Services*, MobiSys '07, pages 139–151, New York, NY, USA, 2007. ACM.

[66] C. Ladha, B.S. Sharif, and C.C. Tsimenidis. Mitigating propagation errors for indoor positioning in wireless sensor networks. In *IEEE International Conference on Mobile Adhoc and Sensor Systems, 2007. MASS 2007*, pages 1–6, October 2007.

[67] Koen Langendoen and Niels Reijers. Distributed localization in wireless sensor networks: a quantitative comparison. *Computer Networks*, 43(4):499–518, November 2003.

[68] S. Lanzisera, D.T. Lin, and K.S.J. Pister. RF Time of Flight Ranging for Wireless Sensor Network Localization. In *2006 International Workshop on Intelligent Solutions in Embedded Systems*, pages 1–12, June 2006.

[69] Yuchun Lee. Handwritten Digit Recognition Using K Nearest-Neighbor, Radial-Basis Function, and Backpropagation Neural Networks. *Neural Computation*, 3(3):440–449, September 1991.

[70] Shyi-Ching Liang, LunHao Liao, and Yen-Chun Lee. Localization Algorithm based on Improved Weighted Centroid in Wireless Sensor Networks. *Journal of Networks*, 9(1), January 2014.

[71] Hui Liu, H. Darabi, P. Banerjee, and Jing Liu. Survey of Wireless Indoor Positioning Techniques and Systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(6):1067–1080, November 2007.

[72] T. Liu, M. Carlberg, G. Chen, J. Chen, J. Kua, and A. Zakhor. Indoor localization and visualization using a human-operated backpack system. In *2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–10, September 2010.

[73] Z.M. Livinsa and S. Jayashri. Performance analysis of diverse environment based on RSSI localization algorithms in wsns. In *2013 IEEE Conference on Information Communication Technologies (ICT)*, pages 572–576, April 2013.

[74] Xiaowei Luo, William J. O'Brien, and Christine L. Julien. Comparative evaluation of Received Signal-Strength Index (RSSI) based indoor localization techniques for construction jobsites. *Advanced Engineering Informatics*, 25(2):355–363, April 2011.

[75] Miklos Maroti, Peter Volgyesi, Sebestyen Dora, Branislav Kusy, Andras Nadas, Akos Ledeczi, Gyorgy Balogh, and Karoly Molnar. Radio Interferometric Geolocation. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, SenSys '05, pages 1–12, New York, NY, USA, 2005. ACM.

[76] S. Mazuelas, A. Bahillo, R.M. Lorenzo, P. Fernandez, F.A. Lago, E. Garcia, Juan Blas, and E.J. Abril. Robust Indoor Positioning Provided by Real-Time RSSI Values in Unmodified WLAN Networks. *IEEE Journal of Selected Topics in Signal Processing*, 3(5):821–831, October 2009.

[77] C. McElroy, D. Neirynck, and M. McLaughlin. Comparison of wireless clock synchronization algorithms for indoor location systems. In *2014 IEEE International Conference on Communications Workshops (ICC)*, pages 157–162, June 2014.

[78] Hamid Mehmood and Nitin K. Tripathi. Cascading artificial neural networks optimized by genetic algorithms and integrated with global navigation satellite system to offer accurate ubiquitous positioning in urban environment. *Computers, Environment and Urban Systems*, 37:35–44, January 2013.

[79] CSAIL MIT. The Cricket Indoor Location System: An NMS Project @ MIT CSAIL.

[80] Mittek Mm, Carlson Jd, Mora-Becerra F, Psota Et, and Perez Ld. In-Home Behavioral Monitoring Using Simultaneous Localization and Activity Detection. *Biomedical sciences instrumentation*, 51:289–296, December 2014.

[81] F.J. Mora-Becerra, P. Sathler, J.D. Carlson, E.T. Psota, and L.C. Perez. Characterizing the RF performance of the eZ430-Chronos wrist watch. In *2014 IEEE*

International Conference on Electro/Information Technology (EIT), pages 399–403, June 2014.

[82] M. V. Moreno-Cano, M. A. Zamora-Izquierdo, Jose Santa, and Antonio F. Skarmeta. An indoor localization system based on artificial neural networks and particle filters applied to intelligent buildings. *Neurocomputing*, 122:116–125, December 2013.

[83] C. Nerguizian, S. Belkhous, A. Azzouz, V. Nerguizian, and M. Saad. Mobile robot geolocation with received signal strength (RSS) fingerprinting technique and neural networks. In *2004 IEEE International Conference on Industrial Technology, 2004. IEEE ICIT '04*, volume 3, pages 1183–1185 Vol. 3, December 2004.

[84] Emin Orhan. Particle Filtering, Condensed. Technical report, University of Rochester, August 2012.

[85] Robert J. Orr and Gregory D. Abowd. The Smart Floor: A Mechanism for Natural User Identification and Tracking. In *CHI '00 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '00, pages 275–276, New York, NY, USA, 2000. ACM.

[86] K. Pahlavan, Xinrong Li, and J.-P. Makela. Indoor geolocation science and technology. *IEEE Communications Magazine*, 40(2):112–118, February 2002.

[87] Ambili Thottam Parameswaran, Mohammad Iftekhar Husain, and Shambhu Upadhyaya. Is rssi a reliable parameter in sensor localization algorithms: An experimental study. September 2009.

[88] S.A. Parkison, E.T. Psota, and L.C. Perez. Automated indoor RFID inventorying using a self-guided micro-aerial vehicle. In *2014 IEEE International Conference on Electro/Information Technology (EIT)*, pages 335–340, June 2014.

[89] Shyamal Patel, Hyung Park, Paolo Bonato, Leighton Chan, and Mary Rodgers. A review of wearable sensors and systems with application in rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, 9(1):21, 2012.

[90] N. Patwari, J.N. Ash, S. Kyperountas, A.O. Hero, R.L. Moses, and N.S. Correal. Locating the nodes: cooperative localization in wireless sensor networks. *IEEE Signal Processing Magazine*, 22(4):54–69, July 2005.

[91] Valery A. Petrushin, Gang Wei, and Anatole V. Gershman. Multiple-camera people localization in an indoor environment. *Knowledge and Information Systems*, 10(2):229–241, May 2006.

[92] S. Poujouly, B. Journet, and D. Placko. Digital laser range finder: phase-shift estimation by undersampling technique. In *The 25th Annual Conference of the IEEE Industrial Electronics Society, 1999. IECON '99 Proceedings*, volume 3, pages 1312–1317 vol.3, 1999.

[93] M. Pricone and A. Caracas. A heterogeneous RSSI-based localization system for indoor and outdoor sports activities. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2014 International*, pages 274–280, August 2014.

[94] R. Priwgharm and P. Chemtanomwong. A comparative study on indoor localization based on RSSI measurement in wireless sensor network. In *2011 Eighth International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 1–6, May 2011.

[95] Daniele Puccinelli and Martin Haenggi. Multipath Fading in Wireless Sensor Networks: Measurements and Interpretation. In *Proceedings of the 2006 International Conference on Wireless Communications and Mobile Computing*, IWCMC '06, pages 1039–1044, New York, NY, USA, 2006. ACM.

[96] Michael Quan, Eduardo Navarro, and Benjamin Peuker. Wi-Fi Localization Using RSSI Fingerprinting. Technical report, California Polytechnic State University, San Luis Obispo, CA, January 2010.

[97] A.N. Raghavan, H. Ananthapadmanaban, M.S. Sivamurugan, and B. Ravindran. Accurate mobile robot localization in indoor environments using bluetooth. In *2010 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4391–4396, May 2010.

[98] N. Ravi, P. Shankar, A. Frankel, A. Elgammal, and L. Iftode. Indoor Localization Using Camera Phones. In *7th IEEE Workshop on Mobile Computing Systems and Applications, 2006. WMCSA '06. Proceedings*, pages 49–49, April 2006.

[99] D.N. Rewadkar and C.N. Aher. Vehicle tracking and positioning in GSM network using optimized SVM model. In *2014 2nd International Conference on Current Trends in Engineering and Technology (ICCTET)*, pages 187–191, July 2014.

[100] Branko Ristic and Sanjeev Arulampalam. *Beyond the Kalman Filter: Particle Filters for Tracking Applications.* Artech House, 2004.

[101] A.H. Sayed, A. Tarighat, and N. Khajehnouri. Network-based wireless location: challenges faced in developing techniques for accurate wireless location information. *IEEE Signal Processing Magazine*, 22(4):24–40, July 2005.

[102] Cyrus Shahabi, Mohammad R. Kolahdouzan, and Mehdi Sharifzadeh. A Road Network Embedding Technique for K-Nearest Neighbor Search in Moving Object Databases. *GeoInformatica*, 7(3):255–273, September 2003.

[103] Hong-Bin Shen, Jie Yang, and Kuo-Chen Chou. Fuzzy KNN for predicting membrane protein types from pseudo-amino acid composition. *Journal of Theoretical Biology*, 240(1):9–13, May 2006.

[104] Shaojie Shen, Nathan Michael, and V. Kumar. Autonomous multi-floor indoor navigation with a computationally constrained MAV. In *2011 IEEE International Conference on Robotics and Automation (ICRA)*, pages 20–25, May 2011.

[105] Xingfa Shen, Zhi Wang, Peng Jiang, Ruizhong Lin, and Youxian Sun. Connectivity and RSSI Based Localization Scheme for Wireless Sensor Networks. In De-Shuang Huang, Xiao-Ping Zhang, and Guang-Bin Huang, editors, *Advances in Intelligent Computing*, number 3645 in Lecture Notes in Computer Science, pages 578–587. Springer Berlin Heidelberg, 2005.

[106] Adam Smith, Hari Balakrishnan, Michel Goraczko, and Nissanka Priyantha. Tracking Moving Devices with the Cricket Location System. In *Proceedings of the 2Nd International Conference on Mobile Systems, Applications, and Services*, MobiSys '04, pages 190–202, New York, NY, USA, 2004. ACM.

[107] A. Thaljaoui, T. Val, N. Nasri, and D. Brulin. BLE localization using RSSI measurements and iRingLA. In *2015 IEEE International Conference on Industrial Technology (ICIT)*, pages 2178–2183, March 2015.

[108] Jiannan Tian and Zhan Xu. RSSI localization algorithm based on RBF neural network. In *2012 IEEE 3rd International Conference on Software Engineering and Service Science (ICSESS)*, pages 321–324, June 2012.

[109] Jorge Torres-Solis, Tiago H. Falk, and Tom Chau. *A Review of Indoor Localization Technologies: towards Navigational Assistance for Topographical Disorientation*. InTech, 2010.

[110] Chih-Yung Tsai, Shuo-Yan Chou, and Shih-Wei Lin. Location-Aware Tour Guide Systems in Museum. In Richard Curran, Shuo-Yan Chou, and Amy Trappey, editors, *Collaborative Product and Service Life Cycle Management for a Sustainable World*, Advanced Concurrent Engineering, pages 349–356. Springer London, 2008.

[111] M. Vari and D. Cassioli. mmWaves RSSI indoor network localization. In *2014 IEEE International Conference on Communications Workshops (ICC)*, pages 127–132, June 2014.

[112] M. Vossiek, L. Wiebking, P. Gulden, J. Wieghardt, C. Hoffmann, and P. Heide. Wireless local positioning. *IEEE Microwave Magazine*, 4(4):77–86, December 2003.

[113] Zu-min Wang and Yi Zheng. The Study of the Weighted Centroid Localization Algorithm Based on RSSI. In *2014 International Conference on Wireless Communication and Sensor Network (WCSN)*, pages 276–279, December 2014.

[114] Kilian Q. Weinberger and Lawrence K. Saul. Distance Metric Learning for Large Margin Nearest Neighbor Classification. *J. Mach. Learn. Res.*, 10:207–244, June 2009.

[115] A. Whitney, J. Fessler, J. Parker, and N. Jacobs. Received signal strength indication signature for passive UHF tags. In *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 1183–1187, July 2014.

[116] Lei Yang, Hao Chen, Qimei Cui, Xuan Fu, and Yifan Zhang. Probabilistic-KNN: A Novel Algorithm for Passive Indoor-Localization Scenario. In *Vehicular Technology Conference (VTC Spring), 2015 IEEE 81st*, pages 1–5, May 2015.

[117] Zheng Yang, Zimu Zhou, and Yunhao Liu. From RSSI to CSI: Indoor Localization via Channel Response. *ACM Comput. Surv.*, 46(2):25:1–25:32, December 2013.

[118] Yunchun Zhang, Zhiyi Fang, Ruixue Li, and Wenpeng Hu. The Design and Implementation of a RSSI-Based Localization System. In *5th International Conference on Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09*, pages 1–4, September 2009.

[119] Lianlin Zhao, E.T. Psota, and L.C. Perez. A comparison between UWB and TDOA systems for smart space localization. In *2014 IEEE International Conference on Electro/Information Technology (EIT)*, pages 179–183, June 2014.

[120] J. Zupan and J. Gasteiger. Neural networks: A new method for solving chemical problems or just a passing phase? *Analytica Chimica Acta*, 248(1):1–30, July 1991.