

MULTIMODAL LEARNING AND SINGLE SOURCE WI-FI BASED INDOOR
LOCALIZATION

Thesis

Submitted to

The School of Engineering of the
UNIVERSITY OF DAYTON

In Partial Fulfillment of the Requirements for
The Degree of
Master of Science in Computer Engineering

By

Hongyu Wu

Dayton, Ohio

May, 2020



MULTIMODAL LEARNING AND SINGLE SOURCE WI-FI BASED INDOOR
LOCALIZATION

Name: Wu, Hongyu

APPROVED BY:

Feng Ye, Ph.D.
Advisory Committee Chairman
Assistant Professor, Electrical and
Computer Engineering

Vijayan Asari, Ph.D.
Committee Member
Professor, Electrical and Computer
Engineering

Bradley Ratliff, Ph.D.
Committee Member
Associate Professor, Electrical and
Computer Engineering

Robert J. Wilkens, Ph.D., P.E.
Associate Dean for Research and Innovation
Professor
School of Engineering

Eddy M. Rojas, Ph.D., M.A., P.E.
Dean, School of Engineering

© Copyright by

Hongyu Wu

All rights reserved

2020

ABSTRACT

MULTIMODAL LEARNING AND SINGLE SOURCE WI-FI BASED INDOOR LOCALIZATION

Name: Wu, Hongyu
University of Dayton

Advisor: Dr. Feng Ye

With the rapid development of high speed Internet and Internet of Things (IoT) applications, the demand of indoor localization technology is increasing over years. Well-developed indoor localization technologies can bring significant changes to industries such as health-care, manufacturing, and security, etc. Wi-Fi fingerprint-based indoor localization is becoming more and more popular thanks to pervasive deployment of Wi-Fi access points and low maintenance cost. However, Wi-Fi fingerprint-based method requires a database of collected signal information from all interest points before hand, which means that the process of data preparation is time-consuming and labor-intensive. Meanwhile, due to the dynamic nature of environment, the persistence of localization system is unstable, thus frequent data re-acquisition and model re-modeling are needed. In addition, current Wi-Fi fingerprint-based methods require multiple WiFi sources, which leads to the increasing amount of cost when constructing the localization system. Therefore, to tackle these issues, a multimodal learning and single source Wi-Fi based indoor localization system is proposed. The proposed system contains three components: Firstly, a moving object detection approach is applied for video processing to generate location labels. Secondly, a single-source Wi-Fi based localization model is developed using the collected signal data as well as the autonomously generated location labels. Lastly, a path tracking scheme is

proposed to demonstrate efficacy of the proposed localization model. Computer based simulation results show that the proposed system provides effective solutions to current indoor localization problems.

I dedicate this work to my parents who supported me to pursue a M.S. degree in Computer Engineering at University of Dayton. Their care and support have sustained me throughout my life. I also dedicate this work to Jie Ji for her care and support.

ACKNOWLEDGMENTS

I am most grateful to the members of my committee, Dr. Feng Ye, Dr. Bradley Ratliff, and Dr. Vijayan Asari, for their time, suggestion, and encouragement throughout this project. Special thanks to the chairman of my committee, who is also my supervisor, Dr. Feng Ye, for his guidance and expertise that assisted my research work.

TABLE OF CONTENTS

ABSTRACT	iii
DEDICATION	v
ACKNOWLEDGMENTS	vi
LIST OF FIGURES	ix
LIST OF TABLES	x
CHAPTER I. INTRODUCTION	1
1.1 Background and Problem Statement	1
1.2 Related Work	3
1.2.1 Wi-Fi based indoor localization	3
1.2.2 Computer vision based object detection	4
1.2.3 Multimodal learning process	5
CHAPTER II. MULTIMODAL LEARNING AND SINGLE SOURCE WI-FI BASED INDOOR LOCALIZATION	7
2.1 Overview of Proposed Indoor Localization System	7
2.2 Video based Localization Data Labeling	8
2.3 Single-source Wi-Fi based Localization	8
2.4 LSTM based Path Tracking	8
CHAPTER III. VIDEO BASED LOCATION DATA LABELING	9
3.1 Hardware Setup	9
3.2 Signal Receiver Detection	9
3.3 Receiver Position Estimation	12
CHAPTER IV. SINGLE-SOURCE WI-FI BASED LOCALIZATION	17
4.1 Single-source Wi-Fi Signal Collection	17
4.2 Signal Data Preprocessing	19
4.2.1 Smoothing and Denoising	19
4.2.2 Data Standardization	21
4.2.3 Multi-scale Down-sampling	22
4.3 Parallel Learning Convolutional Long Short Term Memory Network for Localization	24
4.3.1 Preliminaries	24
4.3.2 Proposed PLConv-LSTM Localization Model	25
4.4 Adaptive Thresholded & Stacked Voting Scheme	27
4.5 Evaluation Results of the PLConv-LSTM Localization Model	30
4.5.1 Settings for Evaluation	30
4.5.2 Evaluation Metrics	32
4.5.3 Evaluation Results	32

CHAPTER V. LSTM BASED PATH TRACKING	35
5.1 Dataset for Path Prediction	35
5.2 LSTM based Path Prediction Model	37
5.3 Evaluation Results	37
5.4 Case Study of Single-source based Wi-Fi Localization for Path Tracing	38
CHAPTER VI. CONCLUSION AND FUTURE WORK	41
BIBLIOGRAPHY	43

LIST OF FIGURES

2.1	Overview of proposed indoor localization system.	7
3.1	Self-driven signal receiver.	10
3.2	A sample of labelled training dataset for YOLOv3.	11
3.3	A test result of object detection network.	11
3.4	The binary image with detected edges in a frame of video.	12
3.5	Top-down view of the detection area for localization.	14
3.6	Top-down view of the detection area for localization.	15
4.1	The experimental indoor area.	18
4.2	A sample of collected RSS data by signal receiver.	19
4.3	A comparison of original and smoothed RSS data of section 0.	21
4.4	An summary of signal data preprocessing.	23
4.5	An summary of proposed PLConvLSTM Network.	27
4.6	Grid search for the optimal threshold.	28
4.7	An overview of adaptive thresholded and stacked voting prediction scheme . . .	29
5.1	Simulated paths.	35
5.2	Sample datasets for LSTM model.	36
5.3	An summary of path tracing scheme.	39
5.4	Graphical interface of the path prediction	40

LIST OF TABLES

4.1	Specifications of the proposed PLConv-LSTM.	30
4.2	Specifications of the PLCNN.	31
4.3	Specifications of the PLLSTM.	31
4.4	Performance of designed localization models.	32
4.5	Performance of PLConv-LSTM with optimization.	33
5.1	Specifications of the path prediction model.	37
5.2	Performance of path prediction model.	38

CHAPTER I

INTRODUCTION

1.1 Background and Problem Statement

In the technology driven world today, localization techniques is highly demanded by many location-based applications such as map and navigation services. Global Positioning System (GPS) [1], is one of the most well-known localization techniques that can determine precise postions of people or objects on earth based on satellite signals. However, due to signal attenuation and scatter by roofs and walls, GPS is not suitable for localization purpose regarding an indoor environment. Therefore, a large amount of research on indoor localization technologies have been explored.

Indoor localization is a process of tracking people and devices relative to an indoor environment. The existing indoor localization techniques can be categorized into two branches: signal-wave based and optical-image based localization [2, 3]. The former refers to the procedure of analyzing mixing source signal wave (sound, radio frequency, etc) based on various signal metrics, in order to estimate the location of the object [4, 5, 6]. The latter utilizes infrared technology and image processing techniques to capture the light pulses or intensity, then accomplish the localization task according to the measurements [7, 8, 9]. Comparing to the optical-image based method, the signal-wave based localization is appealing the attention because of its lower equipement cost and higher localization coverage.

Signal-wave based method can be further classified into two main types: log-distance propagation based model and fingerprint based model. The main idea of signal propagation model is to calculate the distance between the wireless transmitter placed in objects and multiple access points (APs) at many reference points (RPs) of known location. Several

common approaches are adopted, such as Time of Arrival (ToA), Time Difference of Arrival (TDoA), AoA (Angle of Arrival) [10, 11, 12]. However, the log-distance propagation based model requires at least three APs to achieve a promising accuracy, and propagation signals always suffer from the issue of non-line-of-sight (NLOS) [13] multi-paths signals which refers to the interference of obstacles between signal transmitter and receiver. Besides, the position error of one or more APs can greatly bring down the accuracy of the result. On the contrary, fingerprint based model utilizes Channel State Information (CSI) or Received Signal Strength Indicator (RSSI) to obtain users' position, and it minimizes the requirement of environment infrastructure which does not require any additional positioning device [14, 15]. The high accuracy, low cost, and low power consumption features of fingerprint based model have been proved by many researchers. Therefore, fingerprint based localization is becoming one of the most hot topics in recent years.

Fingerprinting method is a scene analysis based localization technique which requires the environmental survey to obtain locations from the detection of signal wave [16]. It is comprised of two phases: offline training and online localization. In the offline training phase, for instance, a site survey is conducted to generate a radio map of observed signal measurements from different locations in an indoor environment. Then, the localization system is deployed based on the collected data. When it comes to the online localization phase, the real-time measurements are compared with the radio map to estimate the user location. Wireless Local Area Network (WLAN), also referred to as “Wi-Fi,” is the most commonly used localization technology due to the widespread WiFi enabled smart devices in the world [17]. With the fact that most of smart devices can obtain received signal information from the APs of WLAN, Wi-Fi based localization method provides an efficient localization environment with low cost infrastructure.

Although regarded as a promising method for indoor localization, Wi-Fi fingerprint based localization still poses a few problems: Firstly, building a fingerprint database is a time-consuming and labor-intensive process which involves a large amount of time and money expense. Secondly, the stability of localization system can easily be affected due to the internal complicated situation of signal wave and external imperceptible interference, such as the presence of other objects, the subtle changes in environment, and signal fluctuation. As a result, it can lead to grossly unsatisfactory localization, especially for long-term deployment. In order to maintain the precision of the localization system, frequently re-acquiring the fingerprints and updating the localization system are highly desired. Finally, to achieve a relatively accurate localization, current Wi-Fi fingerprint based localization methods usually rely on multiple Wi-Fi sources which will inevitably bring up the cost of the entire localization system. Therefore, automating the construction and update of fingerprint database, achieving a high accuracy localization, and meanwhile maintaining the cost of localization system, are necessary in order to accommodate the current situation.

1.2 Related Work

1.2.1 Wi-Fi based indoor localization

With using Wi-Fi as the primary source signal wave, a number of algorithms are adopted in the field of indoor localization. Reference [18] proposed a localization algorithm with Multiple Kernel Learning that measure Wi-Fi signal strength of multiple reference nodes for multiple times to construct the classifiers based on multiple kernel learning. Authors in reference [19] utilize K-Nearest Neighbor algorithm to find the k-nearest matches of online RSSI measurements in offline measurements from multiple access points. Then, the reference position of nearest matches are used to estimate objects location. In [20], author

proposed a kernel method to determine objects' location based on a histogram of the RSSI measurements at unknown locations. In addition, A Bayesian based method called Particle filter is deployed on observed signal which uses Bayesian inference to determine the location based on the highest probability in the resulting distribution [21]. Many researches with the state of art deep learning based artificial neural networks (ANN) and its variants are adopt to extract and consolidate features existing in Wi-Fi signal data, and then predicting the location. In [22], Wen proposed an RSSI based indoor localization method using the ANN, and used RFID reference tag to calibrate the algorithm performance. [23] Wang et al. developed a deep learning based probabilistic method using CSI fingerprint of detected signals, to locate the object. Moreover, in reference [24] author proposed a two-dimensional Convolutional Neural Network (CNN) based localization algorithm using CSI. However, existing Wi-Fi based indoor localization methods always come with a high cost due to the demand of multiple accessible network sources and hardware.

1.2.2 Computer vision based object detection

In general, fingerprints of localization model is usually comprised of two components: the signal information and its corresponding receiver position label. Computer vision, has been widely studied over years. As one of most emerging techniques in the field of computer vision, video based object detection is now becoming a fundamental building block for vision-enabled autonomous systems which helps find the objectness in two dimensional images [25]. In recent years, object detection algorithms are mainly based on Convolutional Neural Networks due to its effectiveness and robustness [26]. Current CNN based object detection is divided into two types: type one is called two-stage detection which splits the detection process into proposal-based feature extraction and classification, represented by

Fast-RCNN [27]. However, two-stage detection is subject to the long time consumption of implementation. YOLO (You Only Look Once) is the representation of one-stage detection, which achieves both localization and classification in one single CNN [28]. Different from two-stage detection networks, YOLO detects object by splitting an image into several grids. By applying convolutional operation to extract the latent features in each block of the image [29], the feature map of the YOLO output layer contains bounding box coordinates, the objectness score, and the class score. The advantage of one-stage detection such as YOLO is its simplicity and fast speed. Therefore, one-stage object detection network is popular for formulating autonomous systems that require real-time detection. When it comes to localization fingerprints preparation, computer vision based object detection algorithms are efficient for continuously tracking the receiver position.

1.2.3 Multimodal learning process

Multimodal learning with various components such as image, sensor, and signal fusioned together, has been an essential solution to a wide range of applications and services human tracking, navigation, and robots routing [30, 31, 32]. To date, a few researchers are working on multimodal learning process that helps achieve autonomous indoor localization. In [33], the author proposed a multi-modal positioning system that combines Bluetooth low energy beacons, Wi-Fi access points and the smartphone's inertial measurement unit, and then calibrating the system with a conditional random field with the incorporation of sparse visual information. Author in [34] developed an image-assist indoor localization system utilizing geometric constraints from crowdsourced images, to improve the performance of Wi-Fi fingerprint localization system. Similarly, in [33], Xu proposed a multimodal approach that maps model from image space to physical space, the algorithm of distance, and orientation

measurements to enhance indoor localization. Some efforts are made specific to fingerprint based localization that focus on automatically obtaining the fingerprints through multimodal process. In [35], authors proposed a localization with altered access points (APs) and fingerprint updating system, which achieves an automatic fingerprint database, update with possibly altered APs. Nevertheless, multiple APs are adopted in this research, which increase the cost of localization system, and the construction of the fingerprint database is still not an autonomous process. [36] Wu and Yang proposed an automatic radio map adaptation method by learning an underlying relationship of smartphones collected real-time RSS dependency between different locations. Similarly, the authors of [37] proposed an recursively update radio maps based on crowdsourcing data collection process in an online fashion is adopted and achieves a relatively accurate localization. However, such data collection process may lead to privacy and security issues relative to the clients since it is mainly based on users' personal devices.

In this thesis work, a Wi-Fi based indoor localization system based on single accessible Wi-Fi source is set up to effectively bring down the cost, and the ultimate goal of this research work is to achieve device free by using any existing access points in directed environment. In addition, multimodal learning approach is adopted, which utilizes computer vision techniques for video based autonomous signal receiver position labeling, to support the learning of Wi-Fi based localization system by combining with corresponding collected signal information.

CHAPTER II

MULTIMODAL LEARNING AND SINGLE SOURCE WI-FI BASED INDOOR LOCALIZATION

2.1 Overview of Proposed Indoor Localization System

To address issues stated above, the proposed multimodal learning and single source Wi-Fi based indoor localization system consists of three major components, including a video based localization data labelling section, a single point Wi-Fi based localization section, and a case study of LSTM based path tracking. An overview of proposed indoor localization system is demonstrated in Fig. 2.1.

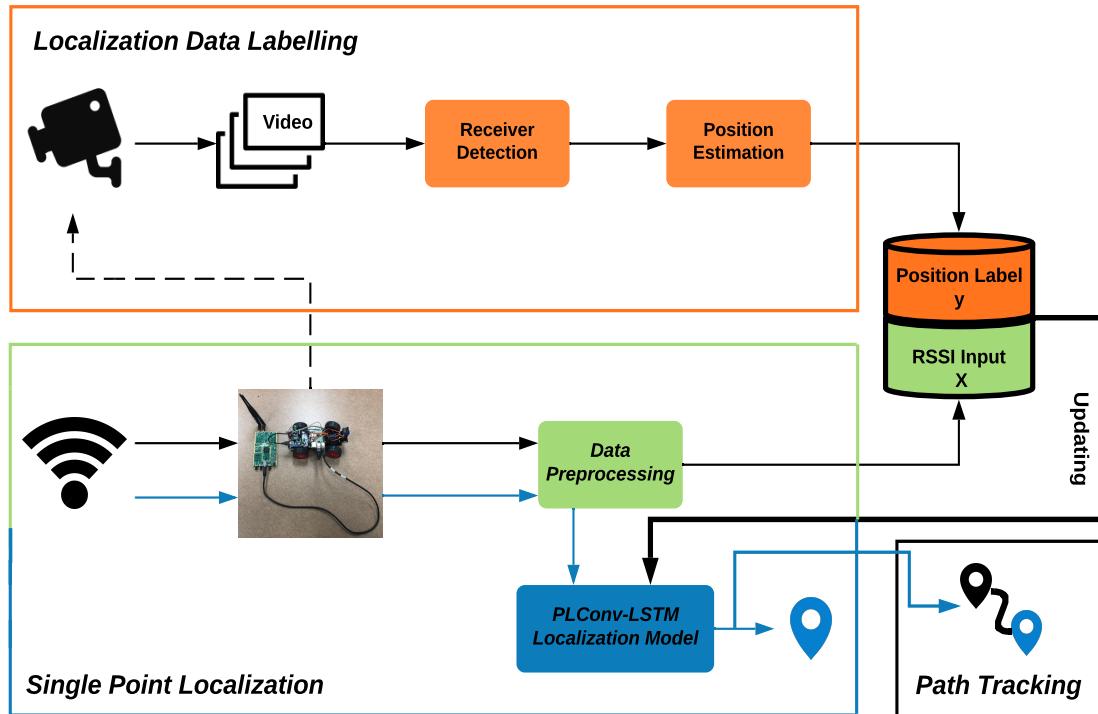


Figure 2.1: Overview of proposed indoor localization system.

2.2 Video based Localization Data Labeling

The video based localization data labelling section involves two parts: Firstly, a computer vision based object detection part that builds a self-driven signal receiver and detects the position of the receiver in a video in real time. Secondly, an autonomous labelling part that transforms the view angle of the camera to a two dimensional view, and estimate the position label based on the real world scale, denoted as y .

2.3 Single-source Wi-Fi based Localization

The single point Wi-Fi based localization section firstly includes preprocessing the collected signal from single Wi-Fi transmitter and deriving the Received Signal Strength (RSS) based input data, denoted as X . By combining X with its corresponding y stated in section 2.2, the offline database for localization model can be constructed and updated autonomously. Moreover, it covers the design of proposed parallel learning Conv-LSTM localization model and comparison of evaluation results on different types of deep learning based localization models. An adaptive thresholded and time sequential voting prediction scheme is also proposed and evaluated.

2.4 LSTM based Path Tracking

Based on proposed indoor localization scheme, a case study of LSTM based path tracking that aims to build a possible indoor navigation application is further researched. An path tracing algorithm is explained and evaluated. The results demonstrate the feasibility of proposed method.

CHAPTER III

VIDEO BASED LOCATION DATA LABELING

3.1 Hardware Setup

In the world of IoT, a microcontroller is a IC chip that executes programs for controlling other devices or machines [38]. Microcontroller boards such as Raspberry Pi and Beagle Bone Black board are popular in embeded software development. To initialize the video based location data labeling section, a self-driven robocar is firstly built by assembling and programming a Raspberry Pi microcontroller board with power-controlled motors, wheels, and a Wi-Fi adapter. In addition, an USRP B200 SDR board is connected to the robocar to perform as a signal receiver. The USRP B200 is a software-defined radio that provides a fully integrated, single board, Universal Software Radio Peripheral platform with continuous frequency coverage from 70 MHz – 6 GHz [39]. By utilizing the constructed self-driven signal receiver, the data preparation process of localization fingerprint database can be achieved automatically without human resources. In an real application scene, instructions will be sent from the console to the self-driven signal receiver which guide the receiver running and collecting signal information in an indoor environment. The working design of the self-driven signal receiver is shown in Fig. 3.1.

3.2 Signal Receiver Detection

In order to obtain the position label of the localization fingerprint database, a computer vision based signal receiver detection is implemented. To determine the objectness of signal receiver in a real-time video, a CNN based one-stage object detection network called YOLO is applied. In addition, YOLOv3, which is the newest version of YOLO, performs greatly

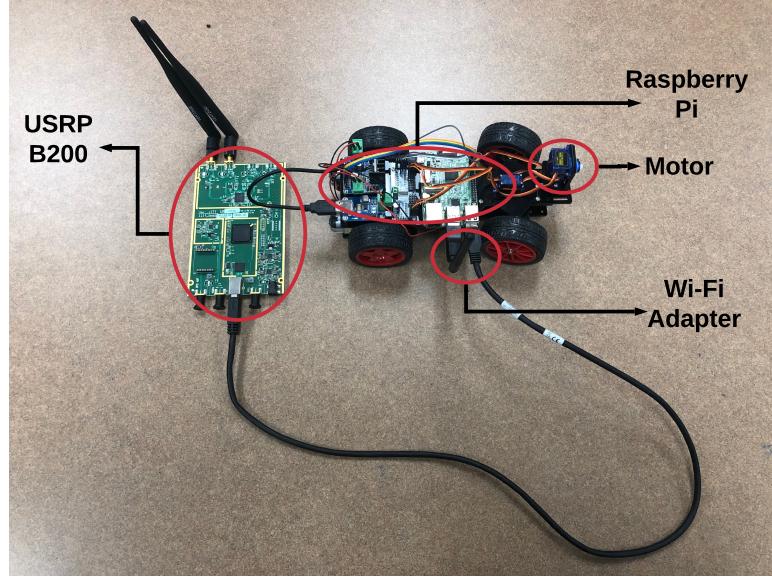


Figure 3.1: Self-driven signal receiver.

even in small object detection and runs in a fast speed without losing its stability [40] comparing to previous versions. Therefore, the video based location data labeling section adapts YOLOv3 as its backbone of computer vision based signal receiver detection.

To accurately detect the self-driven signal receiver existing in an image, a well trained YOLOv3 network is desired. In this research work, hunderds of images contains the signal receiver are taken from different distances and angles. Those images are later labelled as the training dataset for the YOLOv3 object detection network. A sample of labelled training dataset is shown in Fig. 3.2. Besides, a cell phone camera is mounted at a top corner of an indoor environment, to simulate the surveillance camera in real life. A test video is filmed and Fig. 3.3 shows the a frame of test result of the well-trained object detection network.

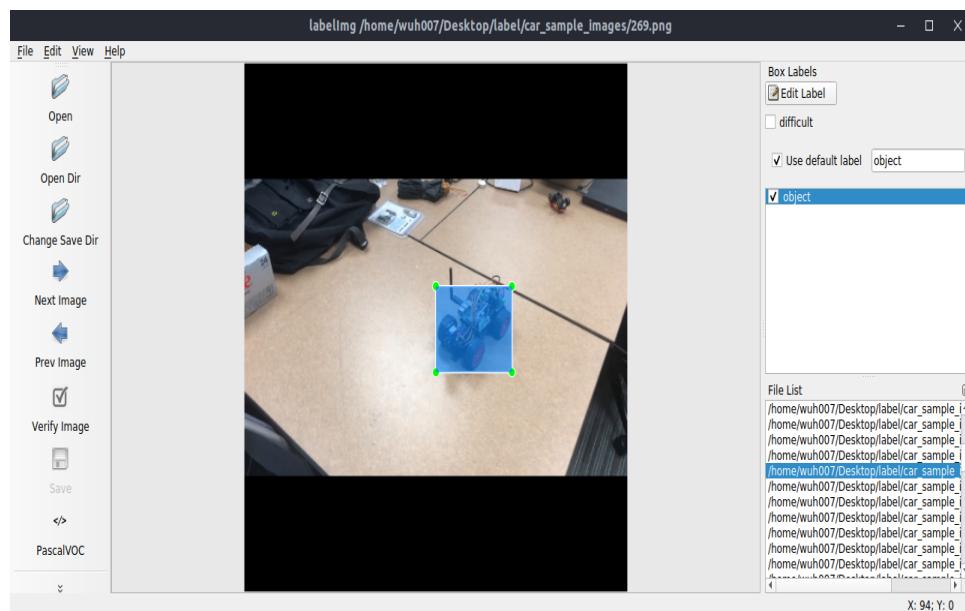


Figure 3.2: A sample of labelled training dataset for YOLOv3.

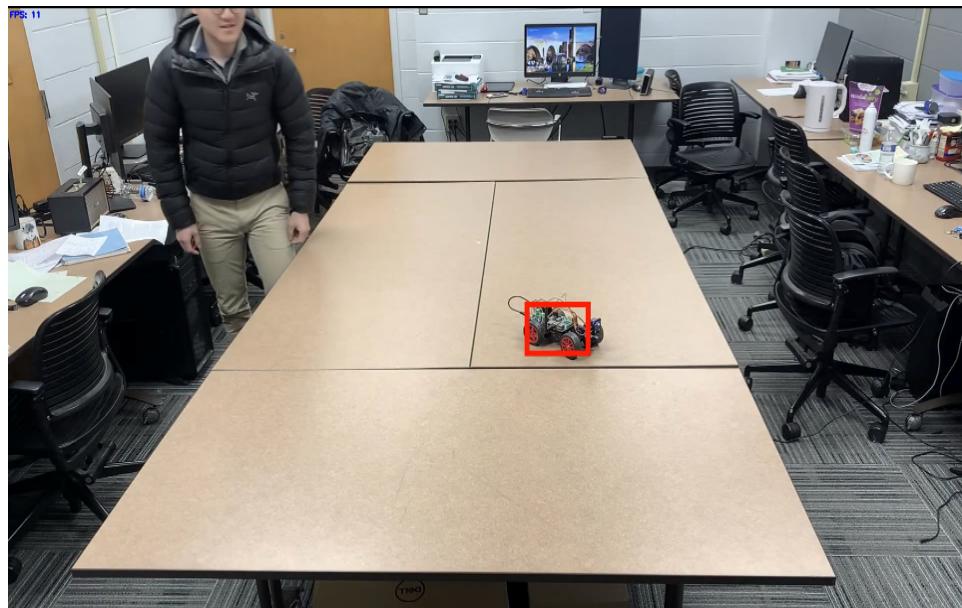


Figure 3.3: A test result of object detection network.

3.3 Receiver Position Estimation

As shown in Fig. 3.3, we assume the table area is the detected indoor area. To automatically find the range of detection indoor area in that frame, a technique called canny edge detection is used [41], which can detect a wide range of edges in images. This technique is comprised of several steps including image noise reduction, gradient calculation, non-maximum suppression, double thresholding, and edge tracking. The biggest area enclosed by detected edges can be therefore computed and selected as the detection area. Fig. 3.4 shows the binary image with detected edges in a frame of video by applying canny edge detection.

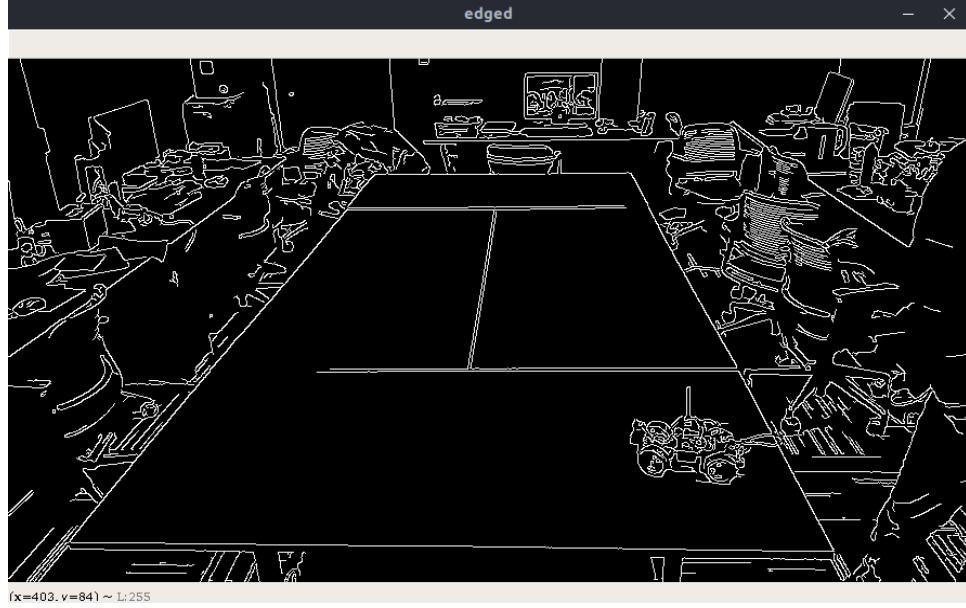


Figure 3.4: The binary image with detected edges in a frame of video.

Benefits from the development of image processing, it is tangible to correct camera perspective to obtain clear and straight view angle of the image. As a result, to tackle the issue of projective distortion due to perspective projection from a 3D scene into a 2D plane

image, the perspective transformation method is adopted. The perspective transformation, also known as homographic mapping, turns the projection of specified area of an image into a new visual plane [42]. The general transformation equation is computed as follows:

$$\begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} \quad (3.1)$$

$$x_2 = \frac{x_1}{z_1} = \frac{h_{11}x_1 + h_{12}y_1 + h_{13}}{h_{31}x_1 + h_{32}y_1 + h_{33}} \quad y_2 = \frac{y_1}{z_1} = \frac{h_{21}x_1 + h_{22}y_1 + h_{23}}{h_{31}x_1 + h_{32}y_1 + h_{33}} \quad (3.2)$$

A 3x3 perspective matrix H is created, which turns a point located in original view plane, denoted as p_1 , into a corresponding point on new view plane, denoted as p_2 . In Eq. 3.1, x_1 and y_1 represent the x-y axis coordinates of p_1 in the original image, corresponding to x_2 and y_2 , which are transformed coordinates of p_2 in new plane. In homogeneous form, the coordinates of this pair of matching points, p_1 and p_2 , are denoted as $(x_1, y_1, z_1)^T$ and $(x_2, y_2, z_2)^T$, without loss of generality. To obtain the inhomogeneous form representation (x_2, y_2) of the coordinates vector $(x_1, y_1, z_1)^T$, Eq. 3.2 is adopted.

By applying canny edge detection and perspective transform to the original image with selected edges of the detected area (called Region of Interest (ROI)), as well as the aspect ratio of such area in real-world coordinator, the top-down, ‘birds-eye view’ of the ROI can be obtained, shown in Fig. 3.5.

Comparing transformed visual plane with the true plane in real-world coordinator, the proportion is maintained. Therefore, the transformed visual plane is considered as a scaled projection of the detection area. Based on the previous two stages: signal receiver detection

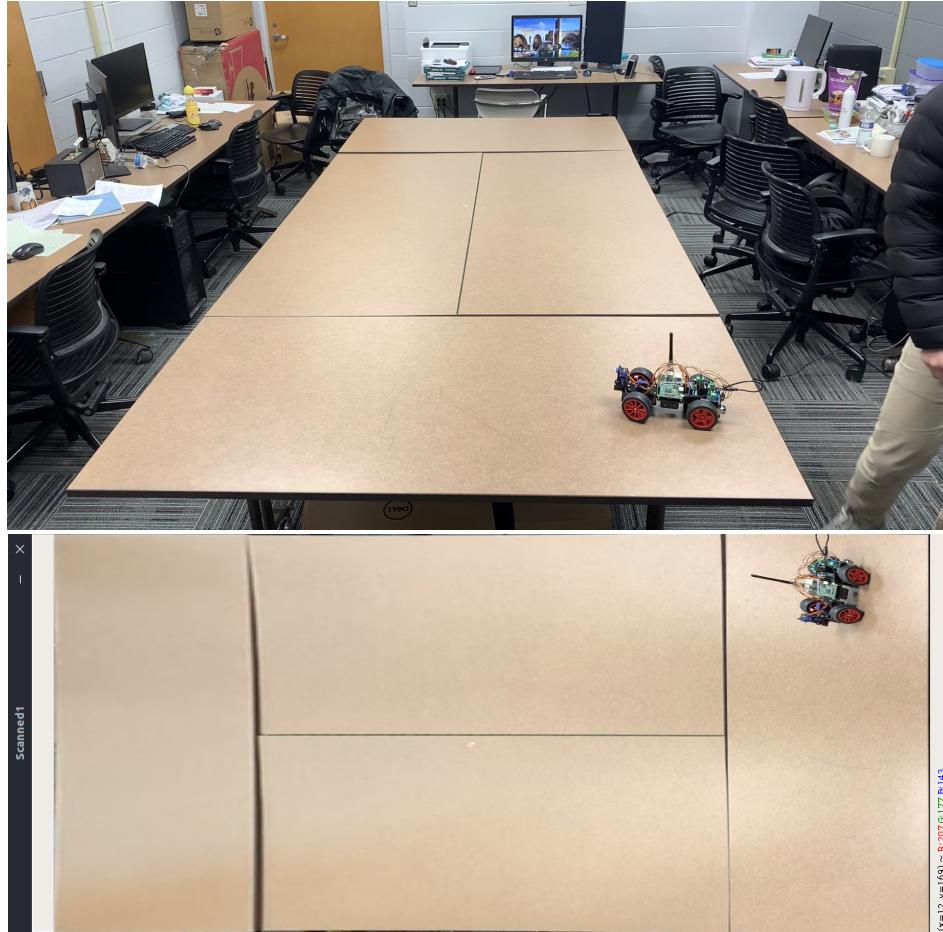


Figure 3.5: Top-down view of the detection area for localization.

and coordinate system transformation, we assume the transformed middle point of detected bounding box in new visual plane as the location of the object or transmitter. Therefore, a new double-polar coordinate system is established with x-y axeses, as shown in Fig. 3.6.

By calculating the ratio of x-y coordinates pixel value regarding to the width and height of the transformed plan, the inverse real location of the detected object can be estimated. The estimation equation is as follows:

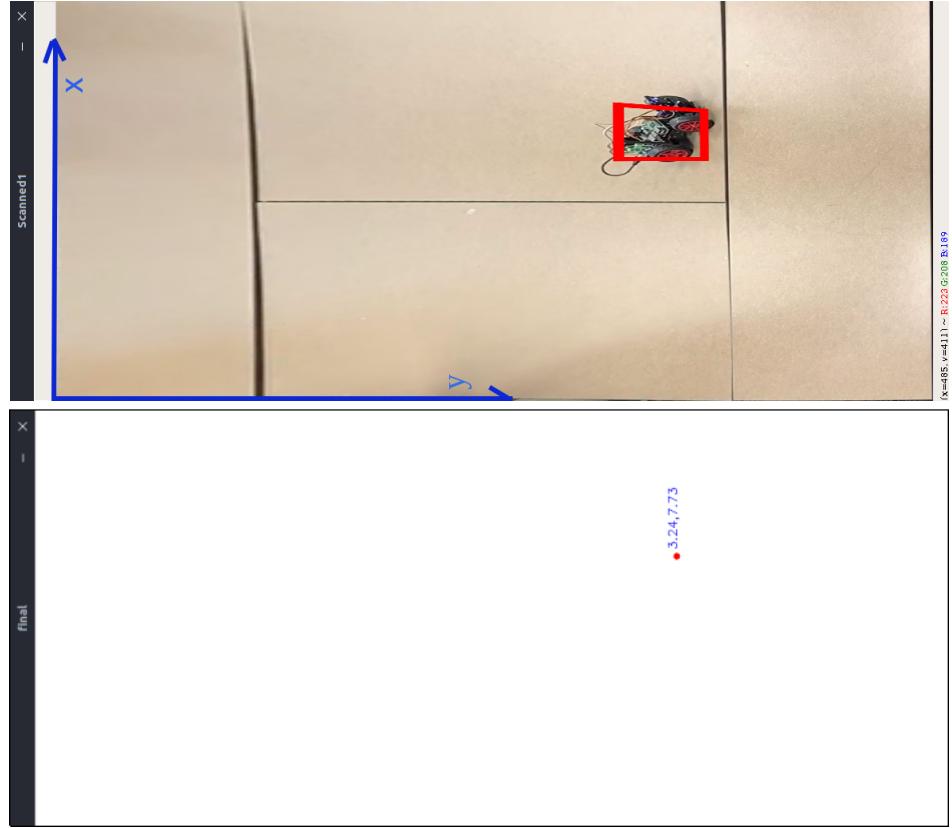


Figure 3.6: Top-down view of the detection area for localization.

$$x'_2 = \frac{x_2}{v} v' \quad y'_2 = \frac{y_2}{u} u' \quad (3.3)$$

With a known point as defined in previous section, denoted as p_2 , we convert it to the point in real-world surface plane, denoted as p'_2 . u and v are the width and height of the transformed plane using perspective transformation. Thus, the real location of point p'_2 with corresponding coordinate vector (x'_2, y'_2) is computed in Eq. 3.3 and the estimated real position of the signal receiver is shown in Fig. 3.6 as well. The estimated position is

converted to position label in next Chapter and used as one significant portion of dataset to establish the localization fingerprint database.

CHAPTER IV

SINGLE-SOURCE WI-FI BASED LOCALIZATION

4.1 Single-source Wi-Fi Signal Collection

As stated in Section. 3.3, the experimental indoor area of the research is a $5 \times 11 \text{ ft}^2$ table. The data collection processs is conducted on the surface of the table. The surface area is firstly divided into 18 sections with an equal size. The estimated receiver position is then converted into the section label. The self-driven signal receiver built in Section. 3.1 is placed on the table to collect signal data. Another USRP B200 board is placed at a fixed place of the experimental area to perform as transmitter which enhance the signal feature in the room. The trasmitting center frequency is set to 5.17 GHz , with a bandwidth of 20 MHz , to simulate a 5 GHz Wi-Fi band. The transmitter is set to keep propogating the signal to surrounding environment. Meanwhile, for the receiver, the receiving frequency range is set to the correspoding center frequency (5.17 GHz) and bandwidth (20 MHz), in order to measure the entire spectrum of the transmitting signal. The fingerprint dataset for localization model contains the measured signal information and its corresponding receiver position label. To obtain enough fingerprints from the experimental area, the self-driven signal receiver is set to traverse all sections and stay for around 3 minutes at each section. In addition, without loss of generality, two groups of data are measured on two different days. The real research area and research are separated to 18 sections as shown in Fig. 4.1.

The working mechanism of the signal receiver is defined as follows: the receiver is imple-
menting a sepecific spectrum analyzer, it does the frequency sweep over required frequency
range and measures the received signal strength of every step. In other word, the receiver
gets N time samples which are sampled at different frequency over the frequency range.

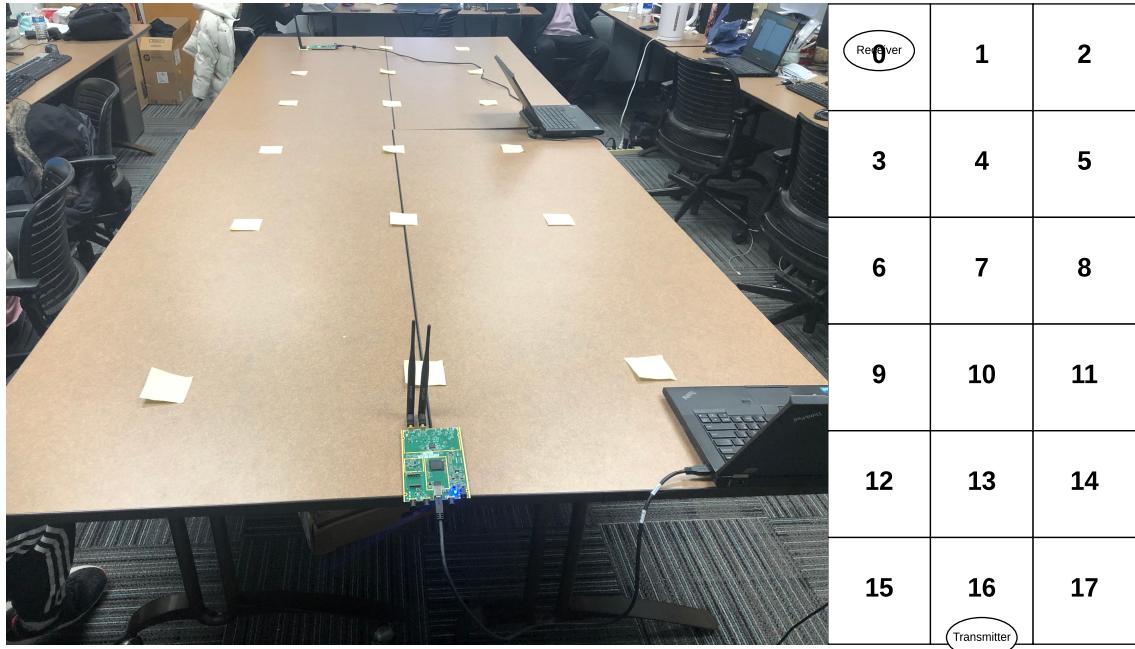


Figure 4.1: The experimental indoor area.

Moreover, the sampling steps are evenly distributed where the frequency and time interval between every two samples are identical.

In this research work, each set of data contains the received signal strength at 144 time steps across the spectrum that is being measured. That is saying, 144 measurements are taken at different frequency step between $5.17 \text{ GHz} \pm 10 \text{ MHz}$, with a same frequency and time interval. In addition, the signal receiver is able to collect at least 5 sets of data per second. For each section in experimental area, 940 datasets are measured in around 3 minutes, and a total of 16922 datasets are obtained from 18 sections. A sample of collected RSS data by signal receiver is shown in Fig. 4.2.

Measure Time	Frequency	RSS
2020-01-27 18:15:39.958634	5160399000	3.78531667770989
2020-01-27 18:15:39.958890	5160532000	6.05733874031705
2020-01-27 18:15:39.959060	5160665000	6.4166192769955
2020-01-27 18:15:39.959225	5160798000	5.93118616498703
2020-01-27 18:15:39.959560	5160931000	8.22502869873267
2020-01-27 18:15:39.959746	5161064000	8.73258005062399
2020-01-27 18:15:39.959953	5161197000	8.14780593838259
2020-01-27 18:15:39.960096	5161330000	5.68022422751761
2020-01-27 18:15:39.960360	5161463000	4.39592533454757
2020-01-27 18:15:39.960511	5161596000	6.80670869954929
2020-01-27 18:15:39.960650	5161729000	5.19245013188112
2020-01-27 18:15:39.960791	5161862000	6.2366340594414
2020-01-27 18:15:39.961025	5161995000	6.9542621619421
2020-01-27 18:15:39.961210	5162128000	6.77123230477534
2020-01-27 18:15:39.961552	5162261000	5.70329472253587
2020-01-27 18:15:39.961751	5162394000	4.37160299891448
2020-01-27 18:15:39.961926	5162527000	8.24278978252042
2020-01-27 18:15:39.962101	5162660000	6.7631155023866
2020-01-27 18:15:39.962236	5162793000	5.36975153287696
2020-01-27 18:15:39.962373	5162926000	4.32248743633051
2020-01-27 18:15:39.962509	5163059000	6.47144607455186
2020-01-27 18:15:39.962652	5163192000	10.4113298230941
2020-01-27 18:15:39.962789	5163325000	13.1784521893009
2020-01-27 18:15:39.962922	5163458000	10.4821170937914
2020-01-27 18:15:39.963057	5163591000	11.3844565372902
2020-01-27 18:15:39.963202	5163724000	11.8919977679753
2020-01-27 18:15:39.963421	5163857000	9.39614775344811
2020-01-27 18:15:39.963556	5163990000	13.0997039116883
2020-01-27 18:15:39.963697	5164123000	12.1839971461046
2020-01-27 18:15:39.963835	5164256000	8.07393662786944
2020-01-27 18:15:39.963967	5164389000	5.22343281379011
2020-01-27 18:15:39.964100	5164522000	5.3632482755645
2020-01-27 18:15:39.964235	5164655000	4.35757739706955
2020-01-27 18:15:39.964372	5164788000	6.25161690801545
2020-01-27 18:15:39.964506	5164921000	5.33027014318913

Figure 4.2: A sample of collected RSS data by signal receiver.

4.2 Signal Data Preprocessing

Data preprocessing is a extremely significant process that allows improving the quality of the raw experimental data. In the area of signal processing, the aim of preprocessing is to remove the outliers, minimize the signal noise, and standardize the data. The section includes smoothing and denoising, data standardization, as well as multi-scale down sampling.

4.2.1 Smoothing and Denoising

Smoothing and Denoising, means removing unwanted spikes, trends and outliers from a signal. In the area of digital signal processing, the smoothing and denoising is the most

common process that needs to be conducted before performing other operations on the signal data. The main concept of signal smoothing is to discover significant patterns in data while leaving out noise, outliers, and other irrelevant information. There are various smoothing and denoising methods existing for preprocssing the signal data, such as Savitzky-Golay filters, moving medians, linear regression, etc. Moving average method, is one of the most well-known filters used by researchers and engineers because of its simplicity to implement [43]. It is an improvmnt over the semiaverage method and it can effectively eliminate the short-term fluctuations of the data. The moving average method can be defined as follows:

Given a set of time series numerical values $Y_1, Y_2, Y_3, \dots, Y_n$ corresponding to time steps $t_1, t_2, t_3, \dots, t_n$, where n indicates the total size of the sampling time steps. Assume the order of moving average is m , it can be calculated by the sequence of arithmetic means:

$$\frac{Y_1 + Y_2 + Y_3 + \dots + Y_m}{m} \quad \frac{Y_2 + Y_3 + \dots + Y_{m+1}}{m} \quad (4.1)$$

In this research work, according to the experimental results, the moving averages of order 5 is selected, which means that the moving average window will compute the average of the beginning five sets of raw data and form the first set of smoothed data. Then the average of second set to sixth set of raw data will be calculated, and so on. The 5 points moving average method effectively reduce the fluctuation in raw data and improve the effectiveness of localization model. A comparison of the collected original RSS data of section 0 and smoothed data shown in Fig. 4.3.

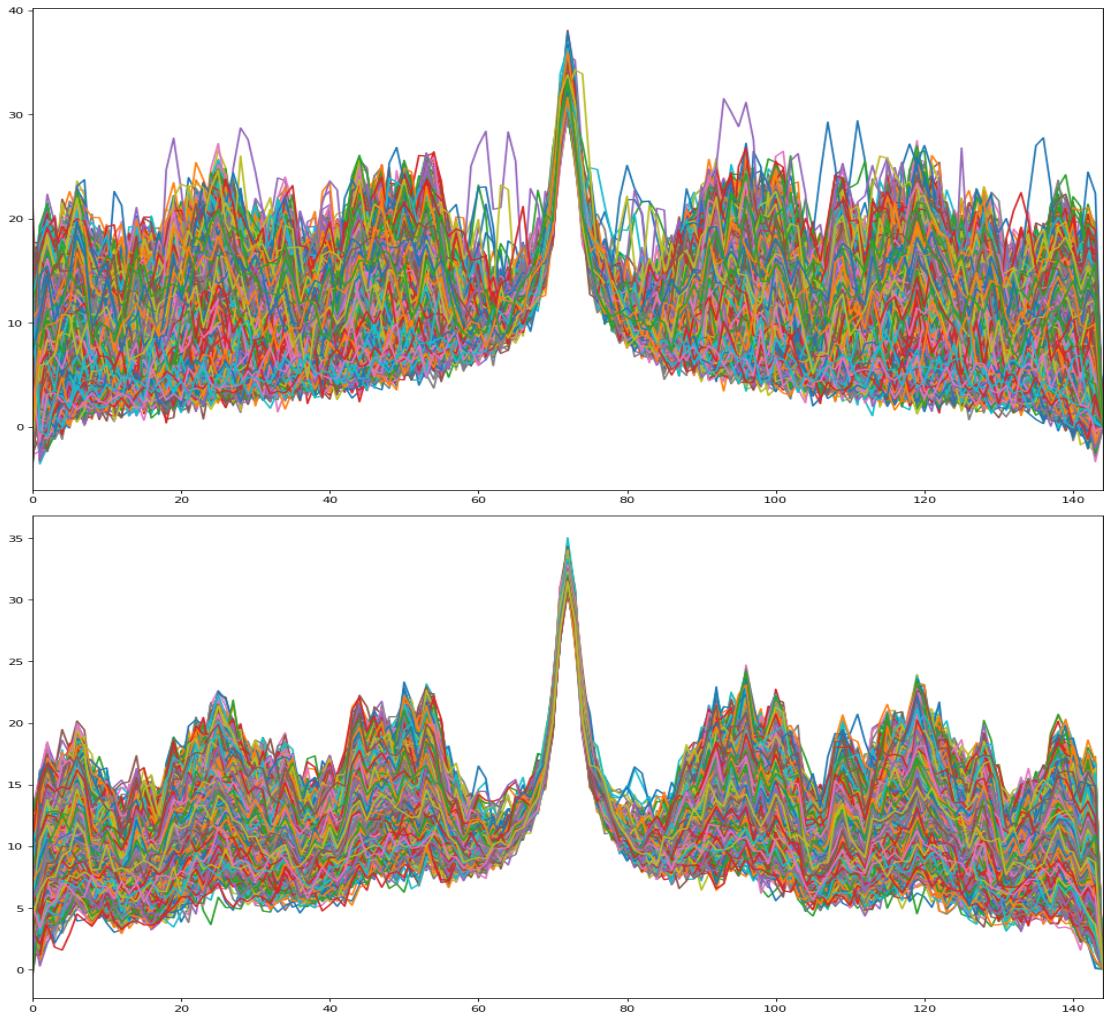


Figure 4.3: A comparison of original and smoothed RSS data of section 0.

4.2.2 Data Standardization

Feature scaling, is a method to normalize the features of data [44]. Although the raw data is initially smoothed and denoised by the method of moving average, the step of data normalization is necessary for feature engineering. In the case of time series signal data, the fluctuation happens frequently due to the effect of surrounding environment. Therefore, if a single value of the feature is orders of magnitude greater than other values of the feature,

that particular value might dominate others in the dataset and affect the performance of localization model.

In this research work, the standard scaler is adopt to transform the data. The standard scaler centers the data by removing the mean value of the feature and then scale it by dividing features by their standard deviation, the equation is as follows:

$$x' = \frac{x - \mu}{\sigma} \quad (4.2)$$

where x' , μ , σ are the final scaled value, the mean and standard deviation of time series data.

4.2.3 Multi-scale Down-sampling

For every time series data, the long-term features reflects overall trends and short-term features depicts more detailed changes in local regions. Times series data in different scales will potentially affects the distinction of features when performing the classification task. Therefore, in order to capture temporal patterns existing in the time series signal data, various down-sampling coefficients are applied to further consolidates information about the time series featrues on a smaller time scale.

In this research work, three different scales are used with a down-sampling coefficient of 0.5, which means that 144 time steps of preprocessed smoothed data are down-scale to 72, 36, and 18, respectively. It is noted that the proposed down-sampling method trims the front and end of the original data, due to the stablity and distinction of features near the center frequency. For example, to achieve 72 time steps, the first and the last 36 time

steps of the original data are removed. An summary of signal data preprocessing is shown in Fig. 4.4.

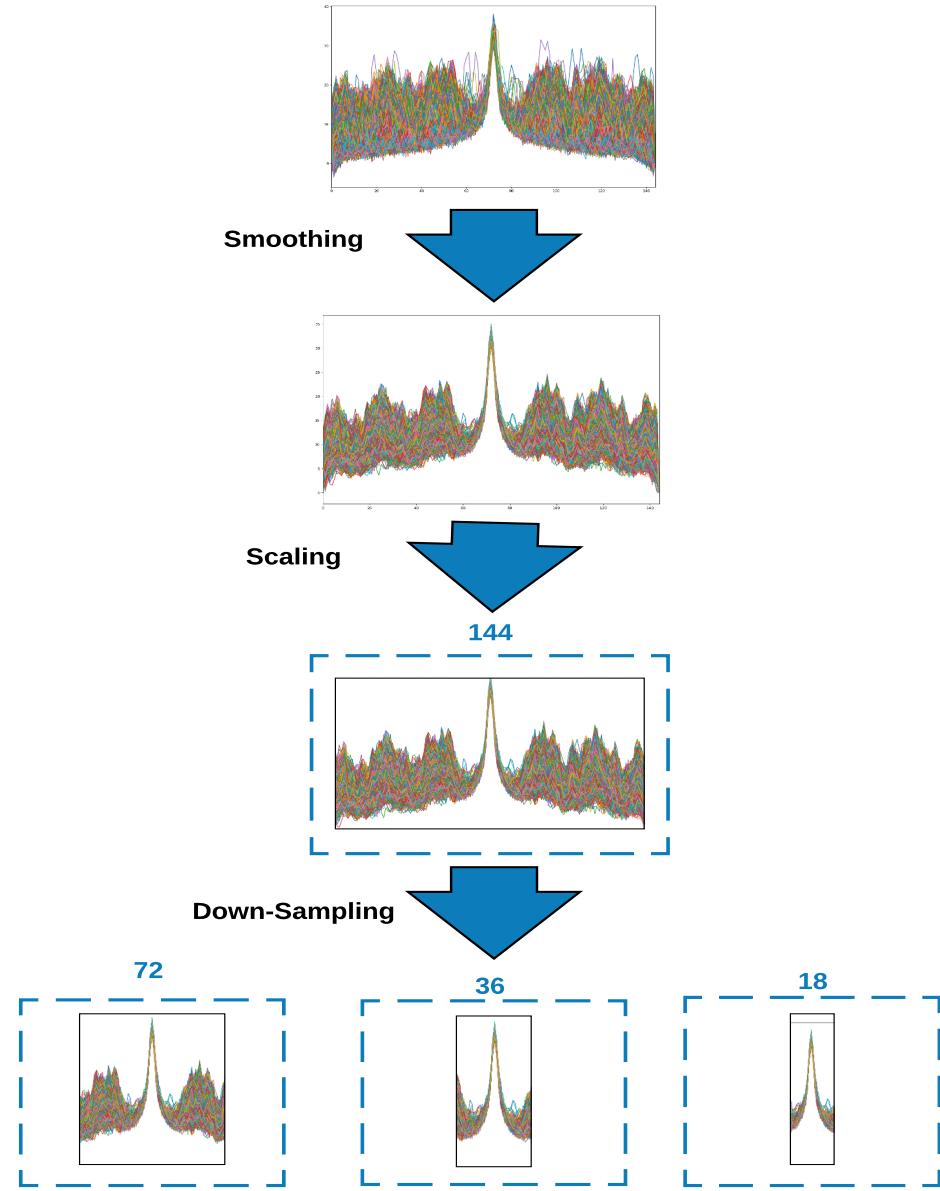


Figure 4.4: An summary of signal data preprocessing.

4.3 Parallel Learning Convolutional Long Short Term Memory Network for Localization

4.3.1 Preliminaries

Recurrent Neural Network is a generalization of feedforward neural network that has an internal memory [45]. RNN performs the recurrent function for every input of data while the output of the current input of data is affected by the last output of the past computation. In other word, in a standard RNN, the decision of every time step depends on the decision made by the previous time step. Therefore, unlike other neural networks, all the inputs in a RNN are related to each other, which makes it a excellent architecture for analyzing time series data. However, RNN often encounters Gradient vanishing and exploding problems due to its short-term memory unit.

To overcome this issue, Long Short Term Memory Recurrent Neural Networks are proposed, which is an improvement over the general RNN by incorporating gating functions into their state dynamics. At each time step, an LSTM computes a mapping from an input x_t to an output o_t by calculating a hidden vector h and a memory vector m responsible for control state updates and outputs. The following computation is performed according to the depict by Graves [46]:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \quad (4.3)$$

$$j_t = \tanh(W_{xm}x_t + W_{hm}h_{t-1} + b_m) \quad (4.4)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \quad (4.5)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \quad (4.6)$$

$$m_t = f_t \otimes m_{t-1} + i_t \otimes j_t \quad (4.7)$$

$$h_t = \tanh(c_t) \otimes o_t \quad (4.8)$$

where i_t , j_t , f_t , o_t are the activation vectors of the input, cell state, forget, and output gates, respectively. $W_h i$, $W_h m$, $W_h f$, $W_h o$ are the recurrent weight matrices and $W_x i$, $W_x m$, $W_x f$, $W_x o$ are their corresponding projection matrices. σ is the logistic sigmoid function and \otimes is an elementwise multiplication operator. Finally, m_t and h_t indicate memory and hidden state at time step t . When a new input comes in, its information will be accumulated to the memory unit if the input gate i_t is activated. In addition, the forget gate f_t controls if the past memory state m_{t-1} needs to be "forgotten". Besides, the output gate o_t will further control if the current memory state m_t will be forwarded to the hidden state h_t .

4.3.2 Proposed PLConv-LSTM Localization Model

Inspired by convolutional LSTM, first presented by Xingjian [47] which extends the traditional LSTM with convolutional layers and is able to extract the spatiotemporal features in time sequence data, a parallel learning convolutional LSTM network is designed for the localization model. Unlike an LSTM that reads the time series data directly in order to calculate internal state and state transition, the convolutional LSTM add convolutional op-

erations directly as part of reading input into the LSTM units themselves. The computation is performed as follows, where $*$ denotes the convolution operator:

$$i_t = \sigma(W_{xi} * x_t + W_{hi} * h_{t-1} + b_i) \quad (4.9)$$

$$j_t = \tanh(W_{xm} * x_t + W_{hm} * h_{t-1} + b_m) \quad (4.10)$$

$$f_t = \sigma(W_{xf} * x_t + W_{hf} * h_{t-1} + b_f) \quad (4.11)$$

$$o_t = \sigma(W_{xo} * x_t + W_{ho} * h_{t-1} + b_o) \quad (4.12)$$

$$m_t = f_t \otimes m_{t-1} + i_t \otimes j_t \quad (4.13)$$

$$h_t = \tanh(c_t) \otimes o_t \quad (4.14)$$

After down sampling, multiple time series with different length from a smoothed time series are generated. In order to feed data into the convolutional LSTM, we reshape each time series into an image like data points. As with a CNN or LSTM network, the output of convolutional LSTM is concatenated and flatten into a long vector that contains all the extracted spatiotemporal features. Those feature are compressed with a dense layer and finally output by the output layer with a Softmax function. The working mechanism of the Softmax function is that it normalizes an input vector into a series of probabilistic values that the input data has probabilities to be any one class among the all, and the sum of those probabilities is 1. For instance, in the case of our localization model, the incoming signal will be classified into one of 18 locations, where the predicted location has the maximum probability value within all locations. An summary of the proposed Parallel Learning Convolutional Long Short Term Memory Network is shown in Fig. 4.5.

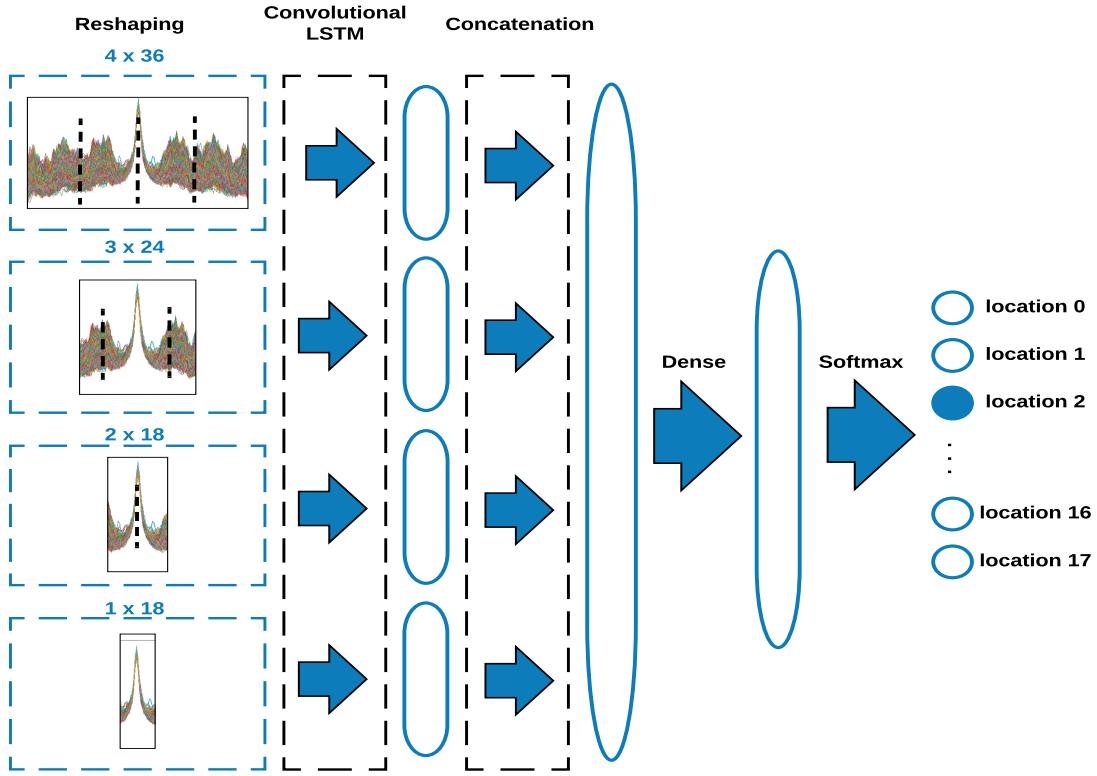


Figure 4.5: An summary of proposed PLConvLSTM Network.

4.4 Adaptive Thresholded & Stacked Voting Scheme

In order to improve localization performance of proposed model, two optimization methods are researched in this section. The Softmax activation function is frequently used in neural networks for classification problems [48]. Assume that the incoming signal goes through the localization model and is correctly classified into location one, with a Softmax output value of 0.65. Even though the prediction is correct, there are more uncertainty associated with an output value of 0.65 versus 0.95. Our previous work [49] is applied, to deal with the uncertainty existing in the localization model using an adaptive threshold is proposed. Firstly, we introduce the drop rate and misclassification rate. The drop rate

indicates the percentage of dropped data regarding to total test data, and the misclassification rate refers to the percentage of filtered misclassified data existing in dropped data. The grid search method is applied, which enumerates the threshold values from 0.5 to 1 with an increment rate of 0.01 and computes the corresponding drop rate and misclassification rate. The grid search result is shown in Fig. 4.6. As we can see, for the test data, at threshold equals to 0.65, the misclassification rate in dropped date reaches the peak, which means most of the dropped data are incorrectly classified by the model, and at this threshold we only drop less than 5 percent test data while improving the performance of the localization model. In addition, the specific threshold value can be varied depending on the localization model.

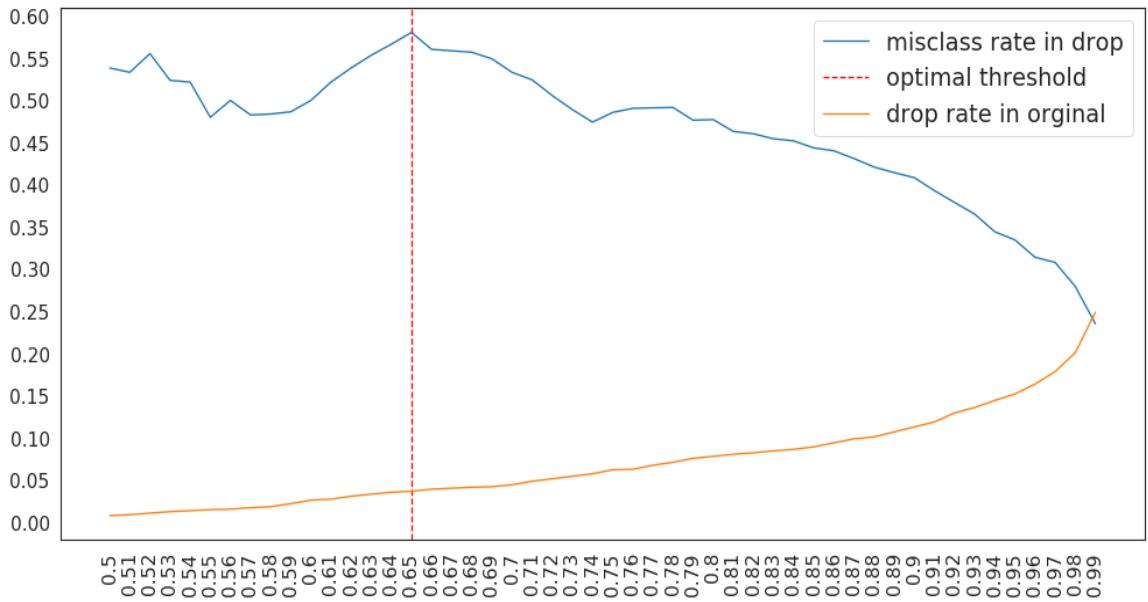


Figure 4.6: Grid search for the optimal threshold.

To further improve the performance of the localization model, a stacked voting strategy is adopted. The proposed strategy is defined as follows: In a real application scenario, at

time t_1 , the incoming signal is received and processed by the signal analyzer, and a location is predicted by the localization model. The prediction of signal is marked as p_1 and is appended into a predefined stack with a specific size. By repeating the previous steps until the stack is full, the localization model will make one concluded prediction based on such: if the size of identical prediction is greater than or equals to 2, the localization system will consider that identical prediction as the current location of the signal receiver, and clear the stack. Otherwise, the stacked predictions will be dropped. In other word, the localization model will make one localization decision every three predictions. An overview of adaptive thresholded and stacked voting approach is shown in Fig. 4.7.

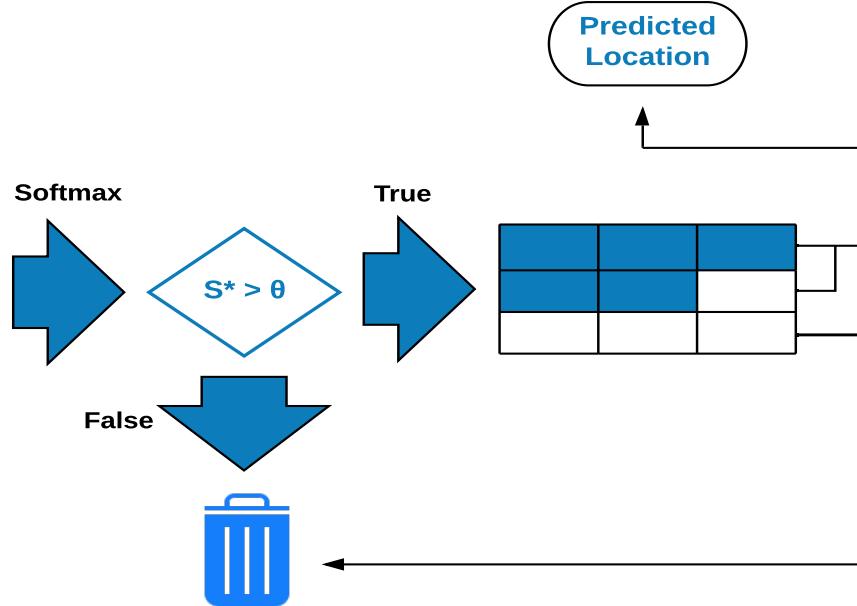


Figure 4.7: An overview of adaptive thresholded and stacked voting prediction scheme

4.5 Evaluation Results of the PLConv-LSTM Localization Model

4.5.1 Settings for Evaluation

By applying preprocessing techniques, four input vector are formed based on the orginal dataset, with corresponding sizes of 144×1 , 72×1 , 36×1 , 18×1 . In order to fullfill input requirements of the convlutional LSTM network, four input vectors are further reshaped, with sizes of $4 \times 1 \times 36 \times 1$, $3 \times 1 \times 24 \times 1$, $2 \times 1 \times 18 \times 1$, and $1 \times 1 \times 18 \times 1$, respectively. For instance, the orginal smoothed input time series data with 144 time steps is divided into four subsequences of 36 time steps. The reconfigured time series has 4 subsequences, 1 row as it is a one-dimensional data, 36 columns as 36 times steps in each subsequence, and 1 feature as the only feature of received signal strength. For the rest of down sampled time series, we reshape them into 3, 2 and 1 subsequences, respectively.

We also evaluate two deep learning based parallel learning localization model by replacing Conv-LSTM with CNN and vanilla LSTM, to compare the performance. Detailed specifications of the built models are listed in Table. 4.1, Table. 4.2, and Table. 4.3.

Table 4.1: Specifications of the proposed PLConv-LSTM.

Classifier type	<i>PLConv-LSTM</i>			
Input size	$4 \times 1 \times 36 \times 1$	$3 \times 1 \times 24 \times 1$	$2 \times 1 \times 18 \times 1$	$1 \times 1 \times 18 \times 1$
Convolutional kernel size	$(1 \times 3) \times 64$			
Activation function	ReLU			
Dropout	0.3			
Flatten	Yes			
Concatenation Parameters	2176	1408	1024	1024
Sizes of dense layers	100 18			
Softmax layer	Yes			

Table 4.2: Specifications of the PLCNN.

Classifier type	<i>PLCNN</i>			
Input size	1×144	1×72	1×36	1×18
Convolutional kernel size	$(1 \times 3) \times 64$ $(1 \times 3) \times 64$			
Padding	Same			
Activation function	ReLU			
MaxPooling	Yes			
Dropout	0.3			
Concatenation Parameters	64	64	64	64
Sizes of dense layers	100 18			
Softmax layer	Yes			

Table 4.3: Specifications of the PLLSTM.

Classifier type	<i>PLLSTM</i>			
Input size	$1 \times 144 \times 1$	$1 \times 72 \times 1$	$1 \times 36 \times 1$	$1 \times 18 \times 1$
Sizes of hidden nodes	144	72	36	18
Return Sequence	Yes			
Activation function	tanh			
Dropout	0.3			
Flatten	Yes			
Concatenation Parameters	20736	5184	1296	324
Sizes of dense layers	100 18			
Softmax layer	Yes			

The evaluation and simulation of the proposed localization model are conducted on a laptop that is equipped with an Intel® Core® i7-9750H CPU @ 2.60GHz, 23.4 GB RAM @ 2133 MHz, 1.4 TB SSD and an NIVIDIA GeForce RTX 2060. Tensorflow 1.13.1 Keras 2.1.5 running in Ubuntu 18.04 is used for the model implementation.

4.5.2 Evaluation Metrics

Accuracy, Recall, Precision and F measurement score are used as shared metrics to evaluate the localization models. In the evaluation of the localization performance, for instance, we define a true positive (TP) decision assigns location 0's signal to location 0, a true negative (TN) decision assigns other locations' signal to other locations. Two types of mistake are committed, for example, one is the false positive (FP) decision, which assigns location 0's signal to other locations. A false negative (FN) decision assigns other locations' signal to location 0.

To evaluate the localization performance, we let TP be the number of true positive instances properly classified as **X**; TN be the number of true negative instances properly classified as **not X**; FP be the number of false positive instances classified as **X** incorrectly; and FN be the number of false negative instances classified as **not X** incorrectly. The evaluation metrics are formulated as follows:

$$R = \frac{TP}{TP + FN}, \quad P = \frac{TP}{TP + FP}, \quad F_\beta = \frac{(\beta^2 + 1)PR}{\beta^2P + R}, \quad (4.15)$$

where $\beta > 1$ can be used as the penalty factor to provide more weight to recall, and $\beta = 1$ is adopted in this thesis.

4.5.3 Evaluation Results

Table 4.4: Performance of designed localization models.

Metrics	Recall	Precision	F1 score	Accuracy	# Training/Testing
PLConv-LSTM	93.2	93.0	93.0	99.3	15230/1692
PLCNN	92.5	92.4	92.4	99.1	15230/1692
PLLSTM	91.2	91.0	91.0	99.0	15230/1692

Table 4.5: Performance of PLConv-LSTM with optimization.

Metrics	Recall	Precision	F1 score	Accuracy	# Prediction
Original	93.2	93.0	93.0	99.3	1692
W. Threshold=0.65	94.8	94.9	94.8	99.4	1602
W. Threshold & S.V.	98.3	98.4	98.3	99.8	534

We split the datasets into 90% training set and 10% testing set, and the performance of localization models are based on the testing results. As shown in Table. 4.4, all of these models achieve excellent results in Accuracy performance. When it comes to other metrics such Recall, Precision and F1 score, the PLLSTM model provides good results but are relatively low comparing to PLCNN model. The reason might be of the data has obscure latent temporal but obvious spatial features. It is also noted that the proposed PLConv-LSTM outperforms baseline models by up to 2% regarding those metrics, which verifies that the extra latent generalized spatio-temporal features are captured by such model. In addition, the performance of PLConv-LSTM with adaptive thresholded and stacked voting prediction strategy is concluded in Table. 4.5. As it can be seen, since the Accuracy performance has already reached a high level, the improvement on Accuracy is not easily perceived. With the optimal pre-defined threshold 0.65, all other metrics increase around 2% and only 90 datasets are dropped, which proves that the adaptive thresholded method can considerably improve the performance of localization model without dropping too much data. In addition, with the involvement of stacked voting method, the Recall, Precision, and F1 score can increase up to 98% based on the optimal threshold, which demonstrates the effectiveness of proposed scheme. However, as a trade-off, the stacked voting method takes a longer time to make one prediction, which may not be suitable for a few scenarios that requires fast speed computation. Overall, according to evaluation results, the designed

PLConv-LSTM model is shown that it is the best model to accomplish the localization task. Also, the adaptive thresholded and stacked voting scheme is proven to boost the accuracy of localization model depending on the performance trade-off.

CHAPTER V

LSTM BASED PATH TRACKING

The ultimate goal of intelligent indoor localization system is to benefit several practical applications such as indoor human navigation and robot routing [50, 51]. Thus, in order to fulfill the proposed localization system, LSTM based path tracking method is proposed to track and predict the movement of an object.

5.1 Dataset for Path Prediction

Without loss of generality, we simulate one forward and one backward paths on the experimental area shown in Fig. 5.1. The process can be extended by adding other paths in a similar way.

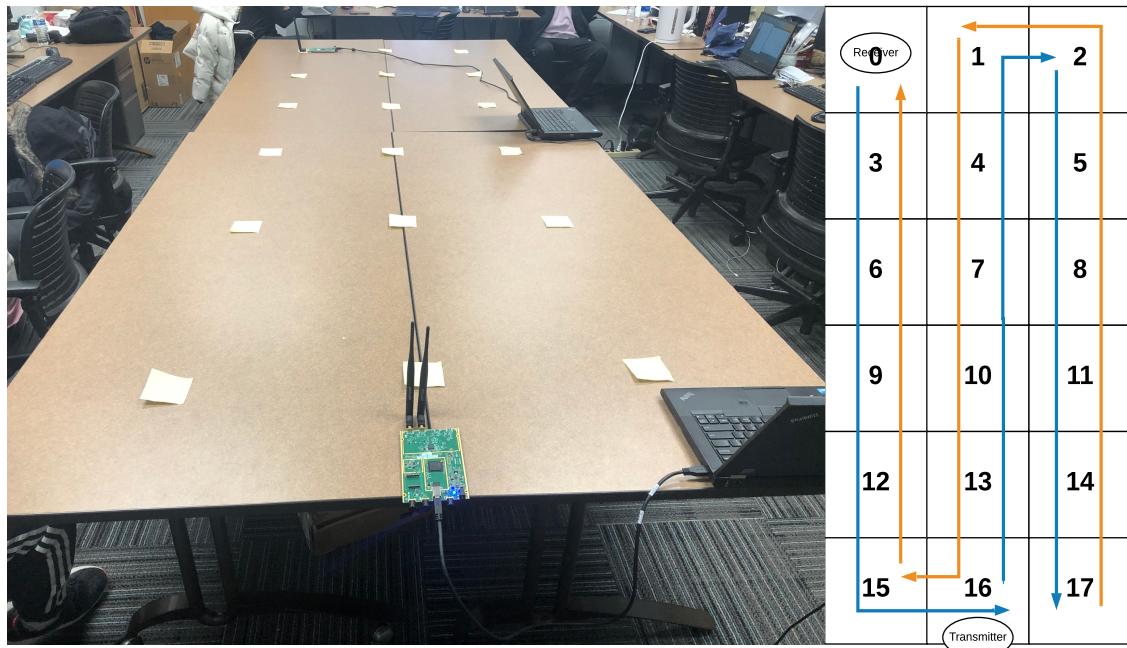


Figure 5.1: Simulated paths.

For each simulated path, we consider the prior three sequential locations of simulated paths as an input sequence and the next location as an output. For instance, in simulated path one, the next location of input sequence ‘3’, ‘6’, ‘9’ is ‘12’. Thus one dataset is formed with input sequence [3, 6, 9] and output [12]. Similarly, for simulated path two, the first dataset can be formed as [17, 14, 11] corresponding to the output [8]. By following the same rule, 32 training datasets are made up iteratively based on the proposed paths, as shown in Fig. 5.2.

Input Sequence	Model Output
[1, 4, 7],	10
[6, 3, 4],	5
[0, 1, 2],	5
[15, 12, 9],	6
[2, 5, 8],	11
[14, 17, 16],	15
[8, 11, 10],	9
[11, 14, 17],	17
[14, 13, 12],	9
[5, 2, 1],	0
[8, 7, 6],	3
[17, 14, 11],	8
[3, 6, 9],	12
[11, 10, 9],	12
[10, 9, 12],	13
[0, 1, 2],	10
[17, 16, 15],	0
[15, 16, 17],	14
[13, 10, 7],	4
[9, 10, 11],	8
[13, 14, 17],	16
[4, 7, 10],	13
[16, 17, 14],	13
[8, 11, 14],	17
[12, 15, 16],	13
[15, 16, 13],	10
[7, 4, 1],	2
[12, 13, 14],	17

Figure 5.2: Sample datasets for LSTM model.

5.2 LSTM based Path Prediction Model

Based on the size of datasets and the characteristic of LSTM networks, the proposed path prediction model uses vanila LSTM as its main structure in this work, the details of the model are concluded in Table. 5.1.

Table 5.1: Specifications of the path prediction model.

Model type	<i>LSTM</i>
Input size	$32 \times 3 \times 1$
Sizes of hidden nodes	100
Return Sequence	No
Activation function	tanh
Sizes of dense layers	1
Activation function	linear

Since the database of prediction model is relatively small, we set the hidden nodes of LSTM network to be 100. One single hidden state for the input sequence with 3 time steps is output by setting the return sequence of LSTM network to ‘False’. The output is activated by ‘tanh’ activation function and feed into the dense layer with an activation function ‘linear’. The constructed time sequential model is able to predict the next location, given the sequence of prior three locations.

5.3 Evaluation Results

In order to evaluate the path prediction model, we consider three different evaluation metrics: Root Sqaure Mean Error (RSME), Mean Absolute Error (MAE), Mean Absolute

Percentage Error (MAPE), and R-squared. The evaluation results are shown in Table. 5.2.

All metrics indicates how well the datasets fit the prediction model.

Table 5.2: Performance of path prediction model.

Metrics	RSME	MAE	MAPE	R-squared	# Datasets
Path prediction model	0.0244	0.0173	0.3700	0.9994	32

As we can see from the table, The RSME and MAE of prediction model are really low, which shows that the prediction error of the model is small. In addition, MAPE value is at 0.37% and R-squared is close to 1, which means that the prediction model reaches perfectness through the training process.

5.4 Case Study of Single-source based Wi-Fi Localization for Path Tracing

By using the LSTM based path prediction model, as a basis, we further introduce a case study of path tracing scheme regarding proposed Wi-Fi based indoor localization system. The scheme is designed to predict the sequential movement of the signal receiver. The study is mainly based on the accuracy of adaptive thresholded prediction since the prediction speed is the first concern when it comes to the path tracing. The scheme for path tracing is described as follows: Similar to the stacked voting method, the localization system will create data structure called “queue”, which has an unique feature called “First In First Out”. For instance, queue is performing as any queue of consumers for a item where the consumer that came first is served first. When it comes to proposed localization system, three predictions are made correponding to first three incoming signals, and they are appended into the queue. Following the rule of stacked voting method, if the size of identical prediction is greater than or equals to 2, the system will output that location. The

difference is that when three predictions are different from each other, they will be used as an input to our pre-trained LSTM based path prediction model. The output of the path prediction model is the predicted next possible location of the signal receiver. When the next signal comes in, the queue will pop out the first or the very left location prediction and append the new prediction to the last or the very right, which ensure smoothing running of the localization system. An summary of path tracing scheme is shown in Fig. 5.3.

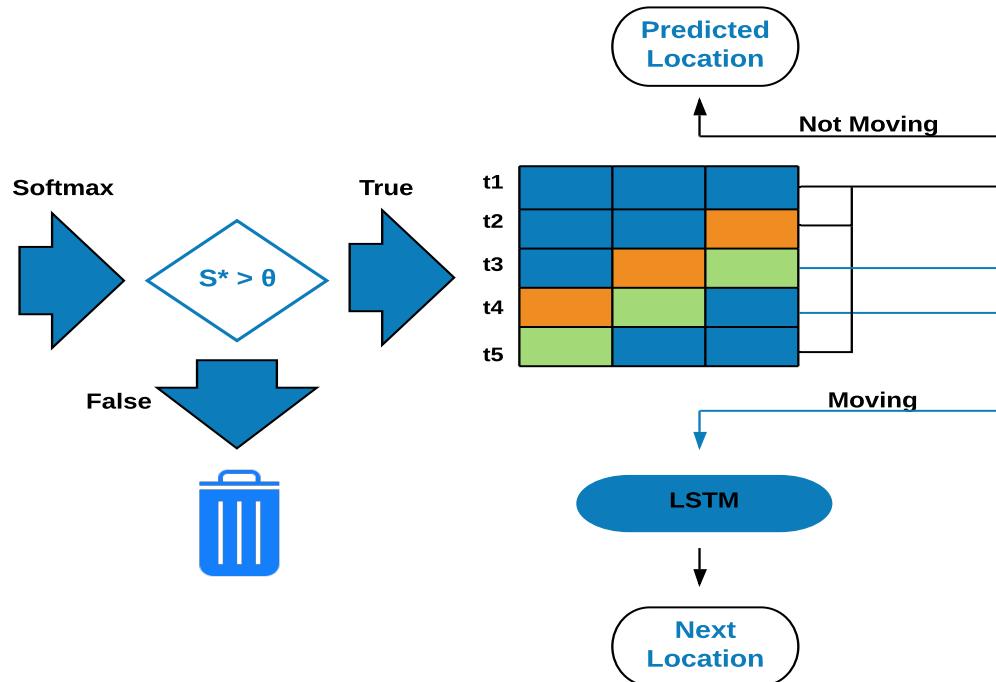


Figure 5.3: An summary of path tracing scheme.

To evaluate the feasibility of the proposed path tracing scheme, we randomly pick dataset from localization fingerprints database that follows the simulated path as discussed in Fig. 5.1. For example, 3 fingerprint datasets at section 3 are picked and follow by 1 dataset at section 6. Then another 2 datasets at section 9 and so on. The selected fin-

gerprint datasets follow the moving route tracking strategy and the output is visualized at each time step. To automate the process of randomly selecting datasets, simulating the path, and visualizing the routing, we develop an application that helps avoid the repeated operations. The graphical interface of the application is shown in Fig. 5.4.

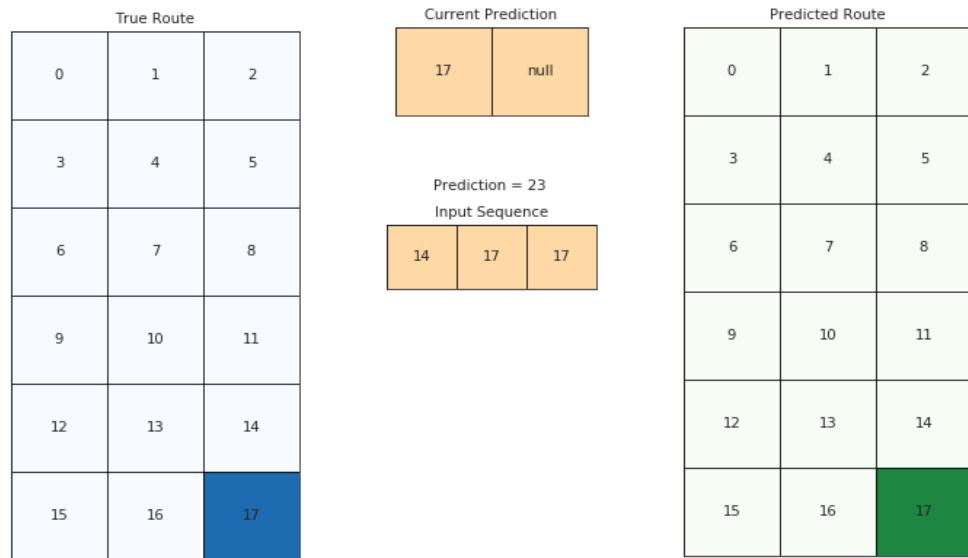


Figure 5.4: Graphical interface of the path prediction

CHAPTER VI

CONCLUSION AND FUTURE WORK

In this paper, we studied the problems of current fingerprint indoor localization technologies. Firstly, we proposed a self-update scheme to automatically update the fingerprint database and localization model with ease. Secondly, we also proposed a single point Wi-Fi source based localization model that effectively classify the signal location while reducing the setup cost to a relatively low level. Lastly, the case study of LSTM based path tracking successfully routing the device within an indoor environment, which is beneficial to the development of intelligent localization and inspires researchers to explore. Although the self-driven signal receiver has not completed yet, the evaluation result prove the feasibility of the autonomous system. In addition, the effectiveness and robustness of proposed localization model has been evaluated.

The proposed Multimodal Learning and Single Source Wi-Fi based Indoor Localization system can efficiently solve the database construction and adapt problems that current indoor localization technologies are encountering, and can perform more effectively and require lower cost than other localization system.

In the future, we will keep working on this area, and some possible goals and improvements are listed as follows:

- Complete the self-driven signal receiver to demonstrate the working system.
- Optimize and improve the current methodology such as position estimation technology to increase the efficiency of current system.
- Add more features to localization fingerprints such as CSI, which may further improve the performance of the system.

- Consider converting the classification problem to regression problem, which may provide more intense and precise localization result.

BIBLIOGRAPHY

- [1] N. Bulusu, J. Heidemann, and D. Estrin, “Gps-less low-cost outdoor localization for very small devices,” *IEEE Personal Communications*, vol. 7, no. 5, pp. 28–34, 2000.
- [2] F. Zafari, A. Gkelias, and K. K. Leung, “A survey of indoor localization systems and technologies,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2568–2599, 2019.
- [3] A. Hameed and H. A. Ahmed, “Survey on indoor positioning applications based on different technologies,” in *2018 12th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS)*, 2018, pp. 1–5.
- [4] R. Machhamer, M. Dziubany, L. Czenkusch, H. Laux, A. Schmeink, K. Gollmer, S. Naumann, and G. Dartmann, “Online offline learning for sound-based indoor localization using low-cost hardware,” *IEEE Access*, vol. 7, pp. 155 088–155 106, 2019.
- [5] E. Hatem, B. El-Hassan, J. Laheurte, S. Abou-Chakra, E. Colin, and C. Marechal, “Study the estimated distance error in indoor localization using uhf-rfid,” in *2018 IEEE Middle East and North Africa Communications Conference (MENACOMM)*, 2018, pp. 1–5.
- [6] H. Zhou-guo, L. Fang, and Y. Yi, “An improved indoor uhf rfid localization method based on deviation correction,” in *2017 4th International Conference on Information Science and Control Engineering (ICISCE)*, July 2017, pp. 1401–1404.
- [7] T. A. Heya, S. E. Arefin, A. Chakrabarty, and M. Alam, “Image processing based indoor localization system for assisting visually impaired people,” in *2018 Ubiquitous Positioning, Indoor Navigation and Location-Based Services (UPINLBS)*, March 2018, pp. 1–7.
- [8] O. Kılınç, G. Küçükyıldız, S. Karakaya, and H. Ocak, “Image processing based indoor localization system,” in *2014 22nd Signal Processing and Communications Applications Conference (SIU)*, April 2014, pp. 1654–1657.
- [9] D. Hauschildt and N. Kirchhof, “Improving indoor position estimation by combining active tdoa ultrasound and passive thermal infrared localization,” in *2011 8th Workshop on Positioning, Navigation and Communication*, 2011, pp. 94–99.
- [10] B. Barua, N. Kandil, N. Hakem, and N. Zaarour, “Indoor localization with uwb and 2.4 ghz bands,” in *2017 IEEE International Symposium on Antennas and Propagation USNC/URSI National Radio Science Meeting*, July 2017, pp. 1403–1404.
- [11] T. Xie, C. Zhang, and Z. Wang, “Wi-fi tdoa indoor localization system based on sdr platform,” in *2017 IEEE International Symposium on Consumer Electronics (ISCE)*, Nov 2017, pp. 82–83.

- [12] H. Chen, B. Hu, L. Zheng, and Z. Wei, “An accurate aoa estimation approach for indoor localization using commodity wi-fi devices,” in *2018 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, Sep. 2018, pp. 1–5.
- [13] M. P. Wylie and J. Holtzman, “The non-line of sight problem in mobile location estimation,” in *Proceedings of ICUPC - 5th International Conference on Universal Personal Communications*, vol. 2, 1996, pp. 827–831 vol.2.
- [14] W. Kui, S. Mao, X. Hei, and F. Li, “Towards accurate indoor localization using channel state information,” in *2018 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, May 2018, pp. 1–2.
- [15] W. K. Zegeye, S. B. Amsalu, Y. Astatke, and F. Moazzami, “Wifi rss fingerprinting indoor localization for mobile devices,” in *2016 IEEE 7th Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)*, Oct 2016, pp. 1–6.
- [16] S. He and S. . G. Chan, “Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons,” *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 466–490, 2016.
- [17] “Ieee standard for information technology—telecommunications and information exchange between systems local and metropolitan area networks—specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications,” *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, pp. 1–3534, Dec 2016.
- [18] H. Fan and Z. Chen, “Wifi based indoor localization with multiple kernel learning,” in *2016 8th IEEE International Conference on Communication Software and Networks (ICCSN)*, June 2016, pp. 474–477.
- [19] A. Belay Adege, Y. Yayeh, G. Berie, H. Lin, L. Yen, and Y. R. Li, “Indoor localization using k-nearest neighbor and artificial neural network back propagation algorithms,” in *2018 27th Wireless and Optical Communication Conference (WOCC)*, April 2018, pp. 1–2.
- [20] C. Huang and H. Manh, “Rss-based indoor positioning based on multi-dimensional kernel modeling and weighted average tracking,” *IEEE Sensors Journal*, vol. 16, no. 9, pp. 3231–3245, May 2016.
- [21] F. Jin, K. Liu, H. Zhang, L. Feng, C. Chen, and W. Wu, “Towards scalable indoor localization with particle filter and wi-fi fingerprint,” in *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, June 2018, pp. 1–2.
- [22] Q. Wen, Y. Liang, C. Wu, A. Tavares, and X. Han, “Indoor localization algorithm based on artificial neural network and radio-frequency identification reference tags,”

Advances in Mechanical Engineering, vol. 10, no. 12, p. 1687814018808682, 2018.
[Online]. Available: <https://doi.org/10.1177/1687814018808682>

- [23] X. Wang, L. Gao, S. Mao, and S. Pandey, “Csi-based fingerprinting for indoor localization: A deep learning approach,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763–776, Jan 2017.
- [24] H. Chen, Y. Zhang, W. Li, X. Tao, and P. Zhang, “Confi: Convolutional neural networks based indoor wi-fi localization using channel state information,” *IEEE Access*, vol. 5, pp. 18 066–18 074, 2017.
- [25] K. R. Reddy, K. H. Priya, and N. Neelima, “Object detection and tracking – a survey,” in *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, 2015, pp. 418–421.
- [26] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [27] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems*, 2015, pp. 91–99.
- [28] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 779–788.
- [29] B.-S. Hua, M.-K. Tran, and S.-K. Yeung, “Pointwise convolutional neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 984–993.
- [30] G. Yang, “Wilocus: Csi based human tracking system in indoor environment,” in *2016 Eighth International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, March 2016, pp. 915–918.
- [31] I. Hwang and Y. J. Jang, “Process mining to discover shoppers’ pathways at a fashion retail store using a wifi-base indoor positioning system,” *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 4, pp. 1786–1792, Oct 2017.
- [32] A. S. Handayani, N. L. Husni, S. Nurmaini, I. Yani, and D. A. Putri, “Mobile-robot positioning for wi-fi signal strength measurement,” in *2017 International Conference on Electrical Engineering and Computer Science (ICECOS)*, Aug 2017, pp. 87–91.
- [33] F. Dümbgen, C. Oeschger, M. Kolundžija, A. Scholefield, E. Girardin, J. Leuenberger, and S. Ayer, “Multi-modal probabilistic indoor localization on a smartphone,” in *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Sep. 2019, pp. 1–8.

- [34] H. Xu, Z. Yang, Z. Zhou, L. Shangguan, K. Yi, and Y. Liu, “Enhancing wifi-based localization with visual clues,” 09 2015, pp. 963–974.
- [35] S. He, W. Lin, and S. . G. Chan, “Indoor localization and automatic fingerprint update with altered ap signals,” *IEEE Transactions on Mobile Computing*, vol. 16, no. 7, pp. 1897–1910, July 2017.
- [36] C. Wu, Z. Yang, and C. Xiao, “Automatic radio map adaptation for indoor localization using smartphones,” *IEEE Transactions on Mobile Computing*, vol. 17, no. 3, pp. 517–528, March 2018.
- [37] B. Huang, Z. Xu, B. Jia, and G. Mao, “An online radio map update scheme for wifi fingerprint-based localization,” *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6909–6918, Aug 2019.
- [38] A. P. James, S. Sugathan, D. A J, and S. S, “An introductory guide to raspberry pi,” 2015.
- [39] J. Talukdar, B. Mehta, K. Aggrawal, and M. Kamani, “Implementation of snr estimation based energy detection on usrp and gnu radio for cognitive radio networks,” in *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, March 2017, pp. 304–308.
- [40] J. Redmon and A. Farhadi, “Yolov3: An incremental improvement. arxiv 2018,” *arXiv preprint arXiv:1804.02767*, 2019.
- [41] J. Canny, “A computational approach to edge detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.
- [42] J. Mezirow, “Perspective transformation,” *Adult Education*, vol. 28, no. 2, pp. 100–110, 1978. [Online]. Available: <https://doi.org/10.1177/074171367802800202>
- [43] S. Smith, *Digital signal processing: a practical guide for engineers and scientists*. Elsevier, 2013.
- [44] A. Zheng and A. Casari, *Feature engineering for machine learning: principles and techniques for data scientists.* ” O’Reilly Media, Inc.”, 2018.
- [45] K. S. Narendra and K. Parthasarathy, “Identification and control of dynamical systems using neural networks,” *IEEE Transactions on Neural Networks*, vol. 1, no. 1, pp. 4–27, 1990.
- [46] A. Graves, A.-r. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013, pp. 6645–6649.
- [47] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, “Convolutional lstm network: A machine learning approach for precipitation nowcasting,” in *Advances in neural information processing systems*, 2015, pp. 802–810.

- [48] M. Sensoy, L. Kaplan, and M. Kandemir, “Evidential deep learning to quantify classification uncertainty,” in *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds. Curran Associates, Inc., 2018, pp. 3179–3189. [Online]. Available: <http://papers.nips.cc/paper/7580-evidential-deep-learning-to-quantify-classification-uncertainty.pdf>
- [49] J. Zhang, F. Li, H. Wu, and F. Ye, “Autonomous model update scheme for deep learning based network traffic classifiers,” in *2019 IEEE Global Communications Conference (GLOBECOM)*, Dec 2019, pp. 1–6.
- [50] H. J. Ju, M. S. Lee, C. G. Park, S. Lee, and S. Park, “Advanced heuristic drift elimination for indoor pedestrian navigation,” in *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2014, pp. 729–732.
- [51] B. Kim, B. Choi, E. Kim, and K. Yang, “Indoor localization using laser scanner and vision marker for intelligent robot,” in *2012 12th International Conference on Control, Automation and Systems*, 2012, pp. 1010–1012.