Transformer-Based Multimodel Indoor Localization Using Wireless and Inertial Sensors

by

Aslan Assylkhanov

Submitted to the Department of Computer Science
in partial fulfillment of the requirements for the degree of

Master of Computer Science

at the

NAZARBAYEV UNIVERSITY

July 2021

Author . . . .
Department of Computer Science
July 16, 2021

Certified by . . . .
Michael Lewis
Associate Professor
Thesis Supervisor

Accepted by . . . .
Vassilios D. Tourassis
Dean, School of Engineering and Digital Sciences

Transformer-Based Multimodel Indoor Localization Using Wireless and Inertial Sensors

by

Aslan Assylkhanov

Submitted to the Department of Computer Science
on July 16, 2021, in partial fulfillment of the
requirements for the degree of
Master of Computer Science

Abstract

The challenge of outdoor navigation has effectively been been met by the deployment of the Global Positioning System (GPS) and the ubiquity of mobile devices equipped with GPS sensors. However, indoor user localization and navigation are less effective , due to problems of device signal attenuation and outright blockage of satellite signals, which in turn hinders critical use-cases such as emergency response and rescue operations. In response to this challenge, researchers have developed a variety of indoor localization algorithms that utilize other methods to approximate location and estimate movement; one example uses the omnipresent wireless network infrastructure, which can be used to determine approximate location by assessing the Received Signal Strength (RSS) data as the user moves about. Using this technique, researchers are steadily improving the accuracy of localization measures.

It is the hypothesis of this work that indoor localization accuracy can be further improved by taking into account other sensors which sample at shorter intervals and higher frequencies. As the proof-of-concept, we utilize Inertial Measurement Unit (IMU) sensor readings in addition to the Wi-Fi Received Signal Strength (RSS). we hypothesize that the IMU measurements can be used to describe the user displacement information and thereby improve the localization accuracy. To test the hypothesis we created an Android mobile application deployed in a testbed environment constrained to a single building, such that we could then collect the necessary RSS and IMU sensor data (from more than 100 path trajectories) needed to test the hypothesis.

The next stage required the development of baseline Recurrent Neural Networks (RNNs) using only the RSS data and then expanding this network to utilize the IMU measurements. Upon verifying the model using our own data, we modified the model to incorporate the use of a state-of-the-art Transformer Networks architecture. Using both the RNN and TN models, we test both RSS data and RSS+IMU data, with the best results attained from the TN model using RSS+IMU data.

Thesis Supervisor: Michael Lewis

Title: Associate Professor

Acknowledgments

This work is the result of my experience as a research assistant at the Institute of Smart Systems and Artificial Intelligence (ISSAI), as part of their localization research group. My work builds on their prior work, and I am grateful for their cooperation with this project, especially under the difficult circumstances of the pandemic restrictions, with the related constraints on laboratory access. At times, I was unable to access the labs, but thanks to their help and comradery my work could continue. I would like to thank my advisors for providing the guidance and support during the research and its documentation process. This has been a work during tough times and I am thankful that I received support from them. I would love to mention and be grateful to my family and friends for their support and being close. Finally, I would like to thank my school and university in general for the opportunities provided and my time spent there. Wish you all the best! May you have pleasant reading, Dear reader!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

"Localization" in the context of communications is characterized by the need to pinpoint the precise location of a user or device within a specific geographic location. One can separate localization services via the environments used, as to whether the user (or device) is located outdoors or indoors. The challenges of outdoor user localization and related navigation services have been largely solved via the development and deployment of the Global Positioning System (GPS) and the ubiquity of GPS sensors in mobile digital devices.

However, the problem of indoor localization remains unresolved, at least with the same precision of location and accuracy of navigation. This lesser performance is attributed to technical complications related to the attenuation of digital device transmissions, relatively slow sampling rates of Wi-Fi signals on mobile devices, uneven coverage by receiving devices, or even the outright blockage of the signals [7].

Current state-of-the-art yields indoor localization precision of about 3.05 meters, which is inadequate for reliable location and navigation services; it could easily be the difference between localization to a room, a hallway, a stairwell, or even open air! (Fig. 1-1)

User localization is recognized as a key component of numerous potential consumer, healthcare, and industrial applications [4], and thus is the subject of significant research interest.

Numerous potential solutions have been proposed to resolve the indoor localization

Figure 1-1: Localization mistakes can be weird

problem [20]. These proposed solutions consider multiple communications technologies (and specialized sensors) such as Wi-Fi signals [2], magnetic field variations [38], ultrasound [6], LED lights [1], ultra-wideband signals [24], cellular signals [11] and their combinations [41], [35].

Amongst these many proposed solutions, the most commonly used are WiFi-based, due to the observation that Wi-Fi infrastructure is widely deployed, therefore is both convenient and practical, and can be readily utilized for proof-of-concept development and potential scalability of solution deployment.

The solutions based on Wi-Fi basically make use of the signal information received from the Wi-Fi Access Points (WAPs). One of the key components of the received information is the Received Signal Strength (RSS) value. The RSS values can be used to create a radio map by using machine learning models to correlate RSS values matched to a Reference Point (RP). Then during the online phase the models attempt to estimate the user's position based on the RSS values received in real-time, as the user or device moves through the environment.

Due to the temporospatial nature of the problem, indoor localization is often addressed using a branch of deep learning known as Recurrent Neural Networks (RNNs). Specifically, the problem is formulated as a sequence problem with regression output learned by the RNNs using the Received Signal Strengths (RSS). However, the wireless sensors of smartphones, computers and other wearable devices often limit the acquisition of the RSS data to intervals of a few seconds, which limits the accuracy of the localization calculations. Presumably, the localization accuracy can be improved by utilizing other inertial sensors such as accelerometers and gyroscopes since this data is sampled at much shorter intervals of 2 milliseconds.

The machine learning localization approaches frequently experience two common hindrances. One is that they typically address the problem as a classification task [19]. Thus, the prediction will be only made as a single point in a classified square. This way there will be no in-between prediction and the refinement of localization requires decreasing the classification square i.e. to have an average accuracy of N meters, one has to divide the testbed into the squares of N x N. However, to increase the accuracy the N has to be smaller, but the smaller the N, the smaller is the area of classification square which in turn decreases by the power of 2 relative to the N decreasing factor. In other words, if N gets decreased by 2, the number of readings need to be done increased by 4; if N get decreased by 3, number of readings increased by 9 and so on. Thus, it requires more extensive data collection process in order to be retain the level of precision.

A second hindrance is that they are made up of multiple separately optimized modules [15], [36], [22], they estimate using only the recent RSS values, ignoring the previous values. Although the good algorithm should consider the following cases, this can cause a very real possibility of having predictions "jump around" the testbed since they are discontinuous.

It should be noted that there are a few solutions that use RNN architecture [42], [21], [26] and also treat the problem in an end-to-end fashion [16].

However, while the end-to-end sequence models provide an improvement, the achieved accuracy of approximately 3.05m is still considered relatively imprecise for purposes of indoor localization and navigation; it is more desirable to attain a precision within one meter.

One of the proposed solutions is the integration of Inertial Measurement Unit (IMU) sensors readings, based on recent published results that show that the models that included consideration of IMU readings yielded better performance than standalone Wi-Fi solutions. Thus, based on this example, we integrate this IMU-augmented approach into the system in order to test this solution strategy using the RNN architecture.

Chapter 2

Literature Review

In this section I will briefly describe the proposed solutions for indoor localization as described in the literature.

Since the indoor localization has been an open problem for more over than a decade and the number of position-based applications are growing over time, there has been a great amount of work done in this field. First, I discuss the data types that are used as an input data for indoor positioning, and then describe multiple algorithms that do the work of position estimation and focus more on the works that are more related to the work of this paper.

2.1 Data Types

2.1.1 RSS

For the purpose of indoor localization, the most useful data is the Received Signal Strength (RSS) data collected as part of network communications from wireless transmitter and receiver sources such as Wi-Fi, Bluetooth, Global Positioning System (GPS), and the Global System for Mobile communications (GSM).

This data can either be used in direct online position estimation or what are known as fingerprint-based solutions.

Direct online position estimation is dependent on the fact that, basically, the distance between the transmitting device and the receiving device can be estimated using the log-normal path loss model where the signal strength is estimated to be logarithmically correlated to the distance between the devices. This information can in turn be used for triangulation of the position. However, the fluctuations of the RSS data in the indoor environment is significant, which serves to diminish the accuracy of the position predictions [3].

As for the fingerprint solutions, they collect the RSS readings during offline phases from devices collected at reference points inside the associated testbed environment. This allows for the construction of a fingerprints database. This database is then used to map the received RSS readings to a probable position during an online-phase. This solution also comes with its own set of drawbacks as the data collection requires a manpower or an optimization process [16], [21] and a stable environment.

2.1.2 Integrated Sensors

Current mobile communication devices are integrated with a number of sensors that can be used to estimate its position. These include inertial sensors (IMU), Magnetometer, Barometers, Light sensors and so on. One of the most commonly integrated and the next prevailing data source is the IMU. The main application it is used with is Pedestrian Dead Reckoning (PDR). What it does is that it provides the direction and distance obtained from the IMU. In general, positioning using PDR requires step length and walking direction at start to track the device/user [14]. Positioning using PDR works best at short distances. However, accuracy may decrease over longer distances since the bias and drift from the sensors accumulate thus, enlarging the error. To compensate for that the PDR data is fused with Wi-Fi data together to achieve better localization accuracy [19], [5]

The barometer is another sensor that can be used in the position estimation. The natural change of pressure based on height can be monitored, and included in the algorithms of indoor positioning [37], [18]. However, the barometer is not generally integrated into modern smartphones, and requires that the environment contain a similar atmosphere, thus are subject to irregularities in indoor environments.

Most mobile devices contain camera sensors that allow for implementation of computer vision techniques to estimate position and navigate the user in the environment. Solutions involving the visual analysis either infer from studying the environment or monitoring of specialized markers whose position is fixed. For studying the environment the camera should be statically placed to analyze the environment [32] thus, the camera that is integrated to the mobile device is not viable for customer navigation-focused applications. Other solutions suggest employing the visual markers for guidance support [9], [29]. Although the solutions involving visual markers tend to be robust, they require the environment to be setup so that the markers themselves are readily available for visual scanning; early adoption of such systems has generally been in warehouse environments to support the automatic navigation of robotic devices.

2.2 Algorithms

As for the algorithms, there are many machine learning solutions for the task of indoor localization. This is expected as machine learning models are useful for retrieving the key features from the data . In the Table 2.1 we have a comparison table for machine learning models grouped by categories. We have provided their strengths, weaknesses, their accuracies and the remarks on their contributions, drawbacks [27][40][10][39][21][42][26][12][16] .

2.3 Related Work

Since my research is dependent on two pillars: employing Wi-Fi RSS signals and Wi-Fi + IMU sensors data; and using machine learning algorithms for extracting the useful information from the data, I will focus on discussing the works that base their work on the similar foundations.

As an example of somewhat earlier usage of machine learning algorithms we can take a look at [27], paper of 2016, where machine learning algorithms where improved

Algorithms	Strenghts	Weaknesses	Accuracy	Comments	Reference
K-Nearest Neighbor(kNN)	Easy to implement; Not responsive to outliers	Requires heavy computations	4.24m	Integration of Principal Component Analysis (PCA) showed improvement in results	[27]
			4.6m	Fingerprting method with RSS; crowdsourced dataset, weighted KNN	[40]
			3.43m	Fuse a group of fingerprints via global fusion profile.	[10]
Support Vector Machines	Provides maping in a non-linear fashion; Ligher with computation requirements; Scalable	Optimization process is complicated; Sensitive to outliers	3.17m	Estimation via kernel-based training method for RSS fingerprinting method	[39]
Neural Network (RNN)	Suitable for sequential data; Arbitrary complex nonlinear function can be approximated	Requires specifically designed and large dataset; Intricate training process	0.75m	Employment of roving robot for data collection; Generated trajectories by random sampling	[21]
			0.86 m	Had generated trajectories using random waypoint model; Incorporated geomagnetic field data;	[42]
			4.92m	Used both real and simulated data; Contained limited number of reference points; implementation using double RNNs	[26]
			2.7m	Comparison of RNN and LSTM solutions. Expansion on the floor detection accuracy up to 98.5%	[12]
			3.05m	Had collected their own dataset; Employed computer vision techniques to decrease the data collection complexity	[16]

Table 2.1: Indoor Positioning Machine Learning Algorithms Comparison Table

by including the Principal Component Analysis (PCA) for feature extraction. This allowed the authors to improve the results of K-Nearest Neighbors (KNN), Linear Support Vector Machine (SVM) and Decision Tree algorithms. This provided an improvement in localization accuracy to 4.24m.

In order to exploit temporal patterns in RSS sequences, [33] created an architecture combining a cascaded Deep Neural Network (DNN) with a Hidden Markov Model (HMM). Specifically, the DNN module uses RSS to estimate coarse positions which are further refined using the HMM module. The DNN module was a four-layer feedforward network with the output layer unit size equal to the number of grid cells in the localization area. The HMM module was employed to estimate finer-level positions and to enforce temporal coherence among them. While this hybrid DNNHMM architecture enables the sequential localization, it is implemented as two distinct modules.

Some works have employed the Convolutional Neural Network (CNN) architectures for the task of indoor positioning: [13] had built a two-dimensional feature map by stacking the sequence of RSS values which is then processed by a CNN network. Even though the approach of using CNN turned out to be effective, it was using a dataset [30] where RSS values were collected arbitrarily, i.e. the RSS values were collected not as a trajectory, but rather statically collected at far located reference points without short-range temporal order.

There are also works that employ RNN machine learning models to solve for the indoor localization issue in a sequential fashion. However, several of them had drawbacks in terms of generating their data, instead of using the realistic ones based on actual trajectories. [26] had used both real and simulated data, but the real data contained a limited number of reference points. [42] had generated uninterrupted trajectories using a random waypoint model to imitate a human walking. In their work they have also incorporated the geomagnetic field data to theoretically improve the estimations. However, as they mention themselves, the two cellphones they used have provided different measurement for magnetic fields although they should have been mostly similar. [21] had exploited the roving robot that had collected the fingerprints

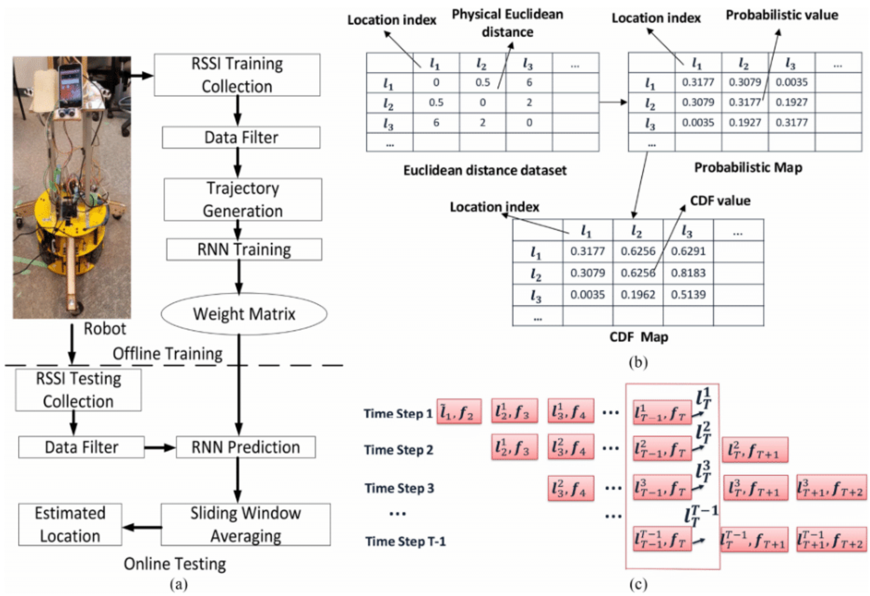
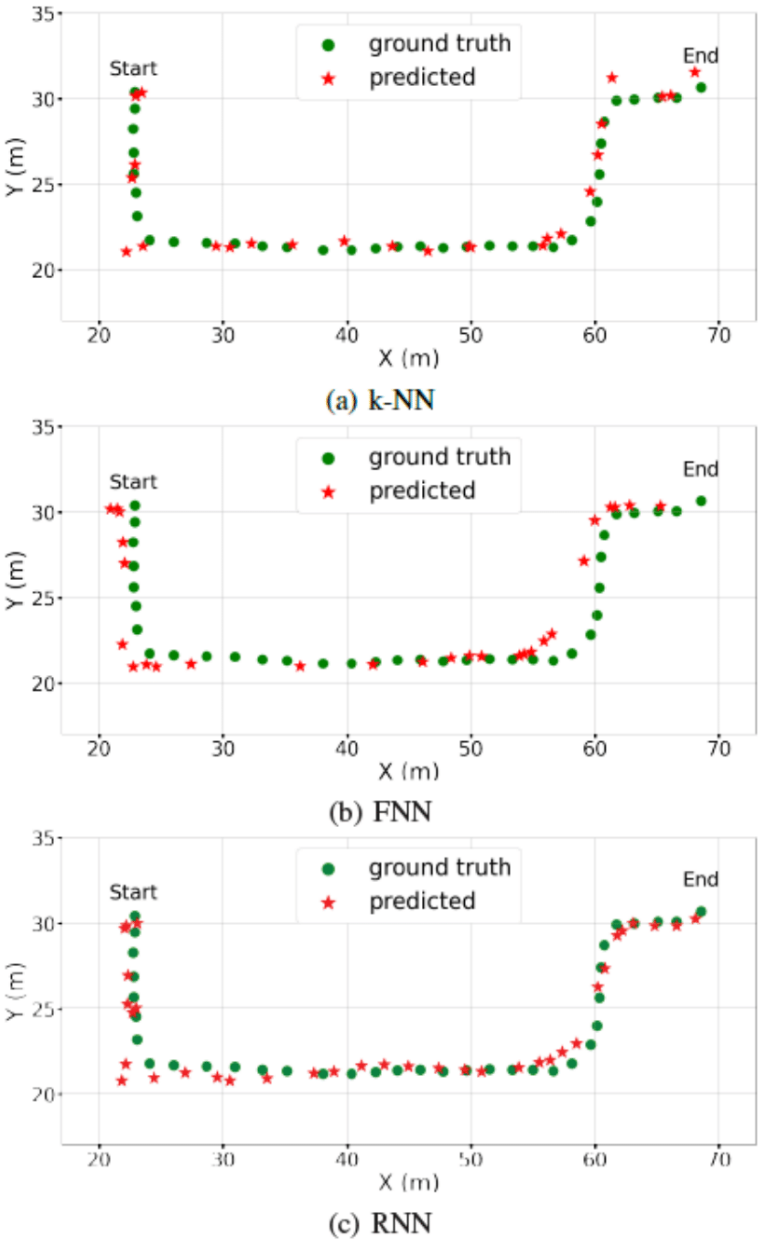


Figure 2-1: An example architecture of indoor localization taken from [21]. a) Localization process of the proposed RNN system. (b) Trajectory generation process. (c) Sliding window averaging in online testing phase.

inside their testbed which is divided into sections. Later, they also generated the trajectories by sampling the data randomly so that a reference point in one section is close to the reference point from the neighbouring sections. Below, you can see the architecture as described by Fig. 2-1, which includes many of the typical elements of this approach, such as the trajectory generation and training in the Recurrent Neural Network.

As another example, [16] had practiced with their own dataset named "WiFine". The dataset was collected in the same testbed as in this work. The trajectories are collected physically with the real displacements across the testbed. Moreover, the dataset has been collected over the several floors of the building making the trajectories more complex. However, in this work they only exploit the Wi-Fi data, without using auxiliary data of other sensors. When compared to algorithms such as KNN, Feedforward Neural Network (FNN) displayed a better accuracy with results of 3.83m, 3.25m and 3.05m respectively. Example of test trajectories the algorithms have displayed can be seen in Fig. 2-2



(a) k-NN

(b) FNN

(c) RNN

Figure 2-2: A test trajectory example with the ground truth and predicted samples using (a) k-NN, (b) FNN and (c) RNN models. The ‘Start’ and ‘End’ text labels are added to indicate the beginning and end of the trajectory. Taken from [16]

Chapter 3

Methodology

In this work the problem of indoor localization is addressed by integrating Inertial Measurement Unit (IMU) sensor data with the Received Signal Strength (RSS) data in transformer networks. We aim to improve the results obtained in [16] as the inclusion of IMU data is expected to improve the localization results by providing the user displacement information at shorter intervals of time.

3.1 Data Collection

This section describes the data collection "testbed" environment which was established by the localization research group of the Institute for Smart Systems and Artificial Intelligence (ISSAI) of Nazarbayev University. In this section, I describe the collection process and outcomes, consisting of over 100 spatiotemporal trajectories of mobile digital devices moving through the testbed pathways.

The trajectories were collected inside the "C4" building of Nazarbayev University, in which the majority of campus research labs are located. The data covers the 4th, 5th and 6th floors including the transition spaces between them, via stairs and elevators (Fig. 3-1). The overall covered area is about 9564 square meters.

The testbed was configured with fiducial markers affixed throughout the designated areas for the purposes of precise calibration of device location in the collection of the trajectory data. For our project we selected ArUco markers, which are popu-

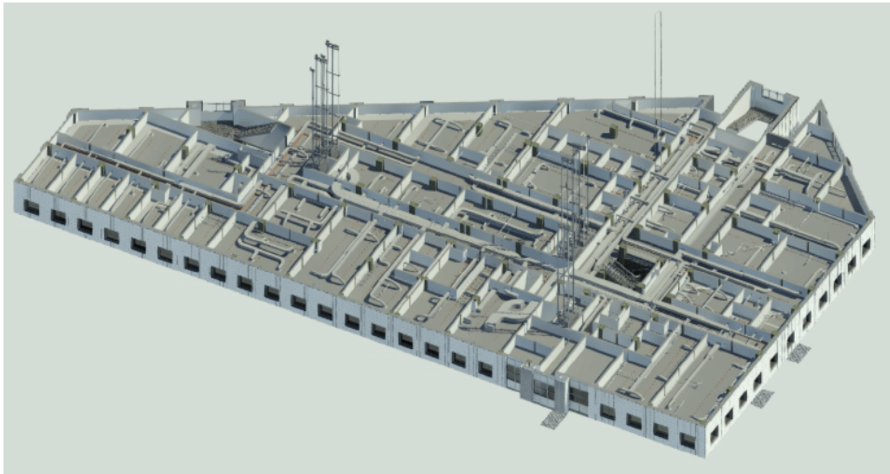


Figure 3-1: The layout of the fourth floor in C4 building.

lar in computer vision and robotics for pose estimation within six degrees-of-freedom [17]. They are comprised of a black 14cm x 14cm square with an inner binary matrix, with patterns generated from the ArUco original dictionary. In total, 654 markers were attached to the walls and the ceilings of the testbed. [16] (Fig. 3-2). Their exact positions were estimated using Leica TS06 plus tachimeter. This estimation allow for precise pose evaluation during the data collection process.

Figure 3-2: ArUco markers attached to the ceilings of the testbed and calibrated

3.2 Data Collection Setup & Process

For this work I have developed a mobile phone application for the Android platform, targeting smart phone devices running under Android 10 (Fig. 3-3).

The platform selection is significant, as in Android 10 the Developer Options allow relatively frequent collection of Received Signal Strength (RSS) data, where the rate is in the range of 2.5 to 6 seconds. The previous version, Android 9, imposed an immutable Wi-Fi throttling feature that limited Wi-fi RSS data collection to only four readings per two minute intervals, which renders it too imprecise for the purposes of indoor localization. Earlier versions, Android 8 and below, allowed more frequent collection, but only for foreground applications. Further, Android 10 remains the most widely-deployed version (even with the release of Android 11), while Android 8 devices are running on fewer than 15% of devices.

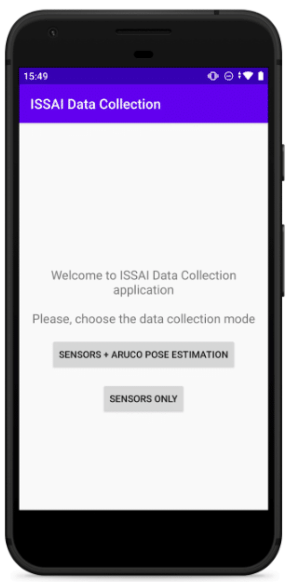


Figure 3-3: Welcome screen of ISSAI Data Collection app

The application was installed on two smartphones: a Samsung Galaxy A21, and an Oppo A5, both running under Android 10.

As the mobile device moves on its trajectory through the testbed environment, it collects the required RSS and IMU data and the corresponding position of the host device. Additionally, the app records the data from other available sensors installed on the device. All of the recorded data is retained, in a .txt file format, for later analysis.

When the app is run the first time, the operator is required to calibrate the camera, using the checkerboard-based method: the user has to take several images of a black and white checkerboard of any desired size (we used 6x8) (Fig. 3-4) at different angles as much as possible. When the image is taken the app process seeks to find the pattern and if one is detected, it notifies the user about it. Overall we tried to collect around 20-30 images for purposes of calibration

.

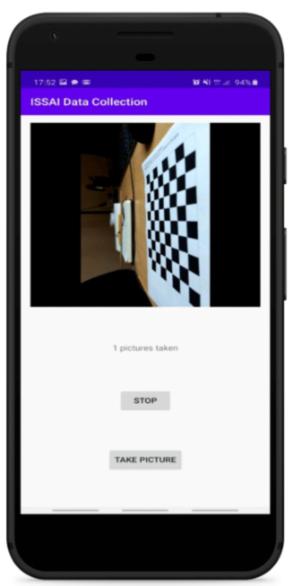


Figure 3-4: Camera calibration process

This is all done to obtain information about camera configuration and establish an accurate information on how the 3D data from the real world is translated into a 2D pixel images that are captured by this camera. This way it allows proper estimation of the user position when detecting the ArUco marker during the data collection.

Additionally, the operator has to choose the desired focus for the camera beforehand. This is done by pressing the focus button while aiming at a specific object (the ArUco marker [8] is preferred), and when the desired focus is achieved we retain the settings. This alignment procedure is conducted prior to each data acquisition process, such that the camera is focused on the same object from the same distance and angle. Same as with camera calibration, this serves the purpose of consistency in data collection and obtaining precise positional data from the ArUco fiducial markers as it sets the camera in the same exact configuration during which the camera calibration was done. The impact of these procedures can be seen in the (Fig. 3-5) and (Fig. 3-6) where one of the paths estimated with uncalibrated data, while the second one is rebuilt from the calibrated and prefocused camera data.

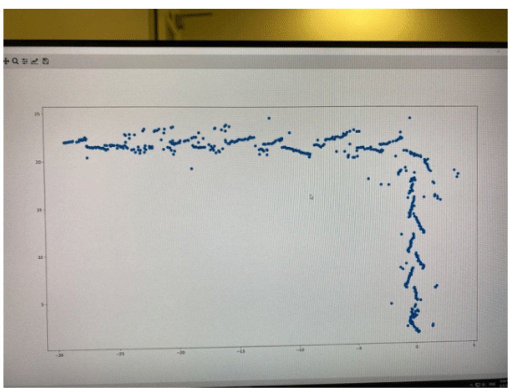


Figure 3-5: Derived path before the camera focus is set

Figure 3-6: Derived path after the camera focus is set

After calibration, the data acquisition process can begin. When the data collection in the app starts the first 30 seconds is spent on a repeated routine of rotating the phone and holding it still for 2-3 seconds, around 8 times altogether. This is done to calibrate the Inertial Measurement Unit sensor reading during the preprocessing [28]. After this step, the operator walks through the building in accordance with the designated trajectory, holding the phone facing forward. At this moment the phone is reading the sensors' data and storing it along with the corresponding timestamps inside the log file.

Additionally, the camera, when it detects the fiducial markers, translates the position (x,y,z) between the marker and the phone to the displacement from the base point of the testbed (Fig. 3-7).

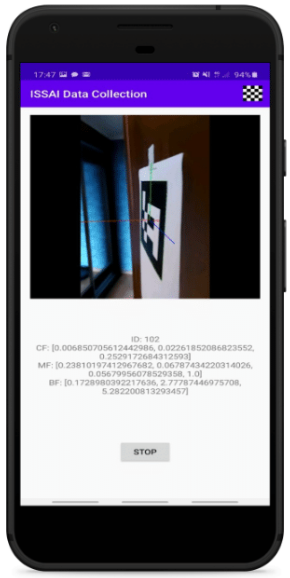


Figure 3-7: Position estimation on ArUco marker

3.3 Dataset Calibration & Preprocessing

After the data is collected it is stored in the corresponding .txt file. Although the file contains all the data needed, it is not easily comprehensible by human reader and is of limited utility to most algorithms as well. This is because the data is added to the file as it is received by the app, chronologically according to the timestamps at which they arrive. They are not otherwise ordered, that is, the data are not aligned with each other, but is represented as just a list of data with respective timestamps. Moreover, the readings from IMU are required to be calibrated in order to regularize its data, to avoid biases from IMU sensors from each phone and retrieve more accurate results.

To resolve all of the issues above we conducted the following preprocessing steps: the data is run through the calibration algorithm [28]. In order to use the algorithm above, in the beginning of every data acquisition process the 30 seconds routine of repeated rotation and pausing movements (as described above) is conducted.

After the data is calibrated it is run through an alignment algorithm which transforms the data from the .txt files into timestamp-aligned data stored in .csv files. To align the data, the data is sorted according to their respective timestamps and also mapped the data to their respective timestamps. Here it is important to remind that the sampling rate for IMU and Wi-Fi are impactfully different: IMU data rate can be set to around 6 or 12ms on the phones used for testing, while the Wi-Fi acquisition rate scatters around 250-300ms.

Considering all of the points above, during the alignment samples of the data are taken each 12ms, and each 300ms the Wi-Fi data is appended. This way our data is consistent with IMU readings and every 25 lines we have the Wi-Fi data. Additionally, since we have 436 Wireless Access Points (WAPs) in the testbed, not all of them can be captured in a single instance and there is a need to somehow map them. Thus, during the alignment, whenever the Wi-Fi data is appended but there is no reading from a certain WAP it is set to the lowest value of -100 while the maximum is 0.

As for the mapping, the unique Basic Service Set IDentifiers (BSSIDs) are stored and mapped to numbers from 0 to 436 and in the alignment column each column is marked as WAP_$wapNumber.

3.4 Data Analytics

3.4.1 Task Definition

The main aim of the indoor localization systems is to estimate the position of either user or a device. In a mathematical form we can describe the position as a vector p that consist of three values x, y, z. Each of these values will represent the position in a horizontal plane (x,y) and across the vertical axis (z). Thus, we can represent it in the following form: p = $(x; y; z) \in R^3$.

In this work we want to estimate the position using the sensor data from the Wi-Fi and IMU data. Given that we have n unique WAPs, the sensors data can be represented in a single vector s = a_x, a_y, a_z, g_x, g_y, g_z, m_x, m_y, m_z, r_1, r_2, ... ,r_n $\in R^{n+9}$ where a_x, a_y, a_z, g_x, g_y, g_z, m_x, m_y, m_z provide information on readings from accelerometer, gyroscope and magnetometer data across the x,y and z axis-es respectively.

Thus, we need to create a function f which takes a vector s as an input parameters and provides an output of a three-dimensional vector $\hat{p}$ which is the position prediction that contains prediction on x,y and z axis-es. This predicted position $\hat{p}$ is then compared to the reference position point associated with that input vector s. Thus, the model can estimate how accurate its prediction is and can learn from it.

Important note. To estimate the position using the RSS values what is usually done is that the data is collected at specific reference points. This allows for a mapping between sensor readings and the position inside the building. Depending on the solution, one might require the position of the WAPs. For example, if the triangulation method is involved, the position is estimated depending on the RSS value and position of the WAP from which the data was sent. However, in the case of this work, machine learning models are used to extract the required features during the training process i.e., as mentioned above, learning from it, such that we do not require the explicit physical location of the wireless access points, rather we infer this information from the data during machine learning.

Moreover, in this work the collected data does not depend on any exact predefined reference point. During the data collection process the reference points are created by estimating the device position via ArUco markers when the other sensor data is collected. Finally, the continuity of the data collected allows us to characterize the problem in a sequence-to-sequence learning fashion. This way the position is estimated taking into account not only the RSS and IMU data of that exact moment, but also the data that was processed before it.

3.4.2 Baseline Solution

Since the current work is an extension of prior work of the ISSAI team [16], we implement their RNN solution as a baseline algorithm to which this algorithm will be compared. This section describes the estimation process done by this baseline solution.

We can separate the baseline solution into its RNN or Hidden Unit estimation and Regression parts as seen in Fig. 3-8. In the Hidden Unit estimation part the algorithm requires the two parameters as an input: sensors data vector s and hidden unit estimation from previous step h_{t-1} thus, working with the data in an orderly fashion. The initial value for h_0 is a zero vector. After the hidden unit vector is estimated it is sent to the regression function that estimates the position by calculating the product of Weight matrix W and summing the result with the bias vector b. Both of the two are learnable parameters.

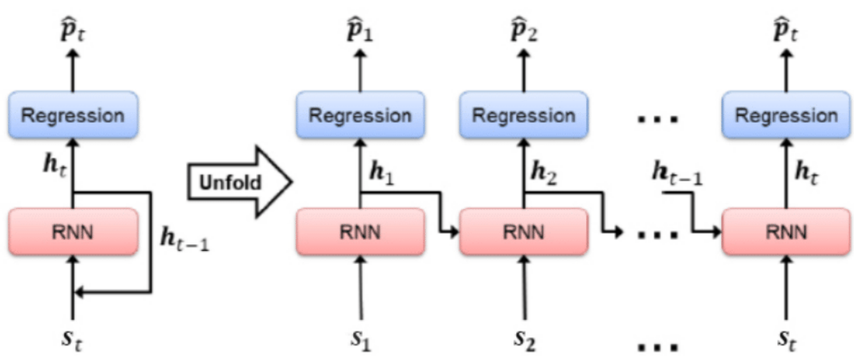


Figure 3-8: Baseline RNN structure

3.4.3 Transformer Neural Network

The structure of the Transformer neural networks is similar to the structure of the baseline RNN solution, except that the transformer models allow for a slight shift in attention change and more parallelization. In this way the model can not only check the data just before it, but it can "pay attention" to the data from various steps before it and can assign them specific weights. Thus, it can look not only to the h_{t-1}, but also for the h_{t-2}, h_{t-3} and so on up to h_{t-n}. This way we can test if such slight focus shifts can improve the results and identify if the transformer models are useful for the purposes of indoor localization.

3.4.4 Training Process & Performance Evaluation

The data was uploaded to a NVIDIA DGX-2 server. The analytics were run on a single GPU (Tesla V100). The system was first run on a baseline Recurrent Neural Network which was separately trained with Wi-Fi data and Wi-Fi + IMU data. Later, the analytics were run on the transformer-based networks also with Wi-Fi data only and Wi-Fi + IMU data to have an indication of whether or not the IMU data improves the results and to what extent. All of the models had 4 prepended rectified linear activation unit (ReLU) layers with 64 hidden units to all models. This was estimated to be the most efficient setup in the previous work from baseline solution [16].

For analysis of the data it was pre-separated into three sections: training, validation and test. All of the trajectories are stored in a separate .csv files detailed in the dataset section. For the training section all of the data is retained to use estimated position information to correct the model predictions, while for the parts of the validation and test we remove those values and just estimate the model from them using the Mean Error Distance (MED) formula as shown in Fig. 3-9. We estimate the training model during using the validation set and estimate the final model using the test set.

$$\text{MED} = \frac{\sum_i dist(\mathbf{p}_i, \hat{\mathbf{p}}_i)}{N}$$

Figure 3-9: Mean Error Distance (MED) formula

Chapter 4

Dataset

As described in the section on Methodology, we have created an Android mobile application that allows us to collect the necessary sensor data and collected 113 trajectories in the C4 research building of Nazarbayev University. All of the trajectories are collected at random, meaning that the person collecting the data could take any route, change direction at any moment, or just stand still. This is done to collect as much diverse data as possible and to imitate the real human walking path. Additionally, the data for training, validation and testing of the algorithm was collected separately at different days. The training dataset covers the whole area of the testbed.

From the 113 collected trajectories 79 were collected using the Samsung A21 phone and 34 were collected using the Oppo A5 phone. The distribution of these trajectories according to the set and the phone can be seen in the Table 4.1

For the data collection we have estimated the upper sampling rate at which all IMU sensors can provide data. The data is then processed with the calibration data collected to remove the bias and then sorted and aligned into a .CSV file.

Set	Samsung Galaxy A21	Oppo A5
Training	58	24
Validation	12	5
Test	9	5

Table 4.1: The distribution of the collected data from different phones across the data sets

Column	Description
1	Timestamp of recorded sample (μs)
2	Acceleration value on X-axis (m/s^2)
3	Acceleration value on Y-axis (m/s^2)
4	Acceleration value on Z-axis (m/s^2)
5	Gyroscope value on X-axis (rad/s)
6	Gyroscope value on Y-axis (rad/s)
7	Gyroscope value on Z-axis (rad/s)
8	Magnetic field value on X-axis (μT)
9	Magnetic field value on Y-axis ((μT)
10	Magnetic field value on Z-axis ((μT)
11	Position on X-axis (m)
12	Position on Y-axis (m)
13	Position on Z-axis (m)
14-449	RSS values of WAPs (dBm)

Table 4.2: Dataset column description

The structure of the resulting .CSV files is represented below in Table 5. Each row represents a sample reading of the data containing all of the required information such as WAPs' RSS values and corresponding IMU sensor reading marked with timestamp that is estimated from the start of the data collection process. All of the rows are organized in a chronological order. As for the quantitative data about the dataset, one can refer to the Table 4.2 and Table 4.3

Dataset	N_B	N_F	N_{RP}	N_S	N_W	N_T	Area
UJIIndorLoc	3	4-5	933	21,049	520	n/a	108,703
XJTLUIndoorLoc	1	2	969	n/a	515	n/a	306
UTSIndoorLoc	1	16	1,840	9,494	589	n/a	44,000
Tampere	1	5	4,648	4,648	991	n/a	22,570
Library	1	2	n/a	63,504	448	n/a	308
JUIndoorLoc	1	3	1,000	25,364	172	n/a	2,646
IPIN2016	4	1-6	2,007	n/a	n/a	26	n/a
IPIN2017	3	1-6	2,697	n/a	n/a	38	n/a
IPIN2018	1	3	n/a	n/a	n/a	38	9,000
WiFine	1	3	26,418	26,418	436	290	9,564
Current work	1	3	5,146	10,283	436	113	9,564

Dataset	Subsets	Total length (m)	Total duration (s)
IPIN 2016	Training	$\approx 8,470$	13,908
	Test	$\approx 4,400$	8,605
	Total	$\approx$ **12,860**	**22,513**
IPIN 2017	Training	$\approx 10,940$	18,411
	Validation	$\approx 3,725$	7,930
	Test	$\approx 3,220$	6,518
	Total	$\approx$ **17,890**	**32,859**
WiFine	Training	$\approx 31,843$	112,330
	Validation	$\approx 4,786$	17,273
	Test	$\approx 6,149$	21,911
	Total	$\approx$ **42,778**	**151,514**
Current work	Training	$\approx 21,628$	76,682
	Validation	$\approx 4,374$	15,849
	Test	$\approx 3,830$	13,133
	Total	$\approx$ **29,832**	**105,664**

Table 4.3: Comparison of dataset statistics

Chapter 5

Results

The thesis distinguished between outdoor and indoor localization, noting that while outdoor localization and related navigation services have largely been solved, there are non-trivial challenges in the context of indoor localization. Improving the accuracy of indoor localization could encourage the development of a suite of consumer- and safety-oriented services similar to those available for outdoor navigation.

The core system architecture was implemented using an approach that utilizes existing Wi-fi infrastructure to generate localization results based on Received Signal Strength (RSS) at the Wi-fi receivers. The experiments were conducted in a testbed environment established within the "C4" research building of Nazarbayev University. Trajectory data was collected from more than 100 routes through the building in order to evaluate the system.

The system was then configured to generate baseline results with the Recurrent Neural Network (RNN) model. As shown in Table 5.1, using the Wi-fi-only method we were able to replicate and slightly improve recently published results, achieving localization accuracy of approximately 2.6 meters.

Having established that the model was functional and generating plausible results, the model was extended to take into account additional sensor data typical of a mobile digital device, with the expectation that the higher frequency and shorter intervals of signaling would allow more precise localization. The Inertial Measurement Unit sensor (IMU), common to most cell phones, was chosen for the proof-of-concept.

The model	Result
RNN (Wi-Fi only)	2.6m
RNN (Wi-Fi + IMU)	2.54m
Transformer (Wi-Fi only)	2.4m
Transformer (Wi-Fi + IMU)	2.3m

Table 5.1: Mean Error Distance(MED) results of different model variations

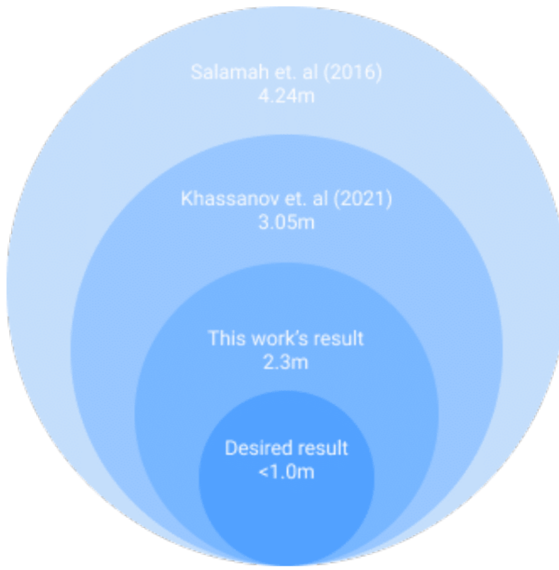


Figure 5-1: Comparison of the localization results over time

For purposes of calibration, the RNN model was used with the Wi-fi + IMU data and achieved localization accuracy results of 2.54 meters, indicating slight improvement over the Wi-fi only model.

The architecture of the system was modified to run on Transformer networks. This modification yielded improved localization accuracy results for both approaches, achieving 2.4 meters for Wi-fi only, and 2.3 meters for Wi-fi + IMU data.

In this thesis, I was able to reproduce a state-of-the art indoor localization system, generate a new data set, and achieve localization results that replicate and slightly improve recently published results. I also demonstrate that the premise of using additional sensor data to achieve more accurate localization results is correct, but the challenge remains as to which sensors and methods can achieve the desired threshold target of one meter localization accuracy.

Chapter 6

Discussion

The work has several objectives, such as collecting the extensive data from Wi-Fi and IMU sensors; employing a new neural network model for the task of indoor localization, replicating previous results and ideally improving on them to achieve the accuracy of 1 meter or less.

Working in the ISSAI laboratory environment, and building on their prior work, the testing system was successfully implemented, and extensive data was collected from the established testbed environment, gathering overall 113 sequential trajectories with total length of approximately 29,832 meters and total duration of 105,564 seconds.

We were able to successfully replicate recently published results [16], and even slightly improve on those results, thus validating the approach. This was achieved in spite of the pandemic restrictions that hindered access to physical facilities and equipment, and introduced substantial delays in the data collection process, the calibration of devices, and the analysis of the data.

However, in this field the goal is to achieve localization of accuracy of one meter or less, while the best obtained results from this work were 2.3 meters. While this result approaches the best results of the Indoor Positioning and Indoor Navigation (IPIN) 2019 results of 1.9 meters [23], where on-site testing was done, it was not as good anticipated during project formulation, and not nearly as good as the as-yet-unexplained results of IPIN 2018 of 0.5m [25]

The transformer neural network was employed for the task of indoor localization. It is perhaps the first time it has been used for the task of indoor positioning, and according to the results it provides a slight improvement. This can probably be explained by the factor of "paying attention" to the previously used data.

Regarding the future work, there are multiple things that I consider can act as an improvement to the work. First of all, I would suggest collecting more data, especially using the Oppo phone or any other phone rather than Samsung Galaxy A21. This is because in our research we had around 70% of the data collected by the Samsung Galaxy A21 phone. This might have been an issue since the Wi-Fi data collected by the two phones tended to differ. Even if the phones are placed in the exact same place and receive the signal at the exact same time, the readings they provide can be distinct. Thus, theoretically, the more abundant data the model has to generalize on, the better it should perform.

Secondly, if we consider the cause of the above-mentioned issue, we can hypothesize that the issue is mostly due to the fact that each phone's Wi-Fi chipset has different gain factor. Thus, the similar signals received can differ after being processed by the chipset. To compensate for that one can implement an algorithm that is designed to infer the gain and can help regularize the readings from different devices. One such solution proposes two solutions using either Gaussian process or neural networks. Both of their solutions provided an improvement to the baseline solutions they have assessed [34]. Hence, I believe that by integrating an algorithm for compensating for the issue of Wi-Fi heterogeneity in either preprocessing or processing part, we can improve the results obtained in this work.

Thirdly, the Wi-Fi data can be preprocessed to have a different representation. In this work the value of undetected WAPs is set to the lowest value. However, there are other Wi-Fi feature representations, and it was found that a suitable representation can produce better results [31]. For example, the ISSAI localization team have continued work in this direction, and independently achieved the accuracy of 1.13m.

Fourthly, we could improve the results by leveraging the building layout as an additional source of information. In the future, we can ask the administration of the testbed building to provide the detailed building layout. This way the information about walls and open spaces can be integrated, thus providing an additional layer of information for the estimation process.

Finally, additional sensor readings can be integrated into the estimation process. For example, some work suggest that using the barometer can provide floor detection with the accuracy up to 99.36% [37]. There is a little hindrance to that the phones used in this work were not equipped with such sensor as barometer is usually found in devices that are more expensive.

Overall, even though the main aim of achieving the localization accuracy of less than 1 meter has not been achieved, I believe that the overall work was successful, overcoming the pandemic circumstances and the shortcomings accompanying it, achieving 3 out of 4 of the stated goals of the project. We have determined that the integration of the inertial sensors can improve the accuracy of localization results. This work also suggests that Transformer Neural Network models can provide a better refinement inthe localization process and can be employed productively in this area.

Bibliography

[1] M. A. Pasha H. Asif T. M. Jadoon A. Naz, N. U. Hassan and C. Yuen. Single led ceiling lamp based indoor positioning system. page "682–687". IEEE World Forum on Internet of Things (WF-IoT), 2018.

[2] C. Basri and A. El Khadimi. Survey on indoor localization system and recent advances of wifi fingerprinting technique. page "253–259". International Conference on Multimedia Computing and Systems (ICMCS), 2016.

[3] R. M. Buehrer and S. A. Zekavat. *Handbook of Position Location: Theory Practice and Advances*. Wiley-IEEE Press, Hoboken, NJ, USA, 2011.

[4] S. Kim S. Lee L. Wirola C. Laoudias, A. Moreira and C. Fischione. A survey of enabling technologies for network localization, tracking, and navigation. *IEEE Communications Surveys Tutorials*, 20(4):"3607–3644", 2018.

[5] Frédéric Evennou and François Marx. Advanced integration of wifi and inertial navigation systems for indoor mobile positioning. *EURASIP J. Appl. Signal Process*, 2006, 12 2006.

[6] D. J. Schott L. M. Reindl F. Hoeflinger, A. Saphala and C. Schindelhauer. Passive indoor-localization using echoes of ultrasound signals. page "60–65". International Conference on Advanced Information Technologies (ICAIT), 2019.

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

[8] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco Madrid-Cuevas, and Manuel Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47:2280–2292, 06 2014.

[9] Jayavardhana Gubbi, N. K. Sandeep, K. Pavan Kumar Reddy, and P. Balamuralidhar. Robust markers for visual navigation using reed-solomon codes. In *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, pages 460–463, 2017.

[10] Xiansheng Guo, Shilin Zhu, Lin Li, Fangzi Hu, and Nirwan Ansari. Accurate wifi localization by unsupervised fusion of extended candidate location set. *IEEE Internet of Things Journal*, 6(2):2476–2485, 2019.

[11] M. Torki H. Rizk and M. Youssef. Cellindeep: Robust and accurate cellular-based indoor localization via deep learning. *IEEE Sensors Journal*, 19:"2305–2312", 2019.

[12] He-Yen Hsieh, Setya Widyawan Prakosa, and Jenq-Shiou Leu. Towards the implementation of recurrent neural network schemes for wifi fingerprint-based indoor positioning. In *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pages 1–5, 2018.

[13] Mai Ibrahim, Marwan Torki, and Mustafa ElNainay. Cnn based indoor localization using rss time-series. In *2018 IEEE Symposium on Computers and Communications (ISCC)*, pages 01044–01049, 2018.

[14] Yunye Jin, Hong-Song Toh, Wee-Seng Soh, and Wai-Choong Wong. A robust dead-reckoning pedestrian tracking system with low cost sensors. In *2011 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 222–230, 2011.

[15] S. Lee K. S. Kim and K. Huang. A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on wifi fingerprinting. *Big Data Analytics*, 3(1):"4", 2018.

[16] Y. Khassanov, M. Nurpeiissov, A. Sarkytbayev, A. Kuzdeuov, and H. A. Varol. Finer-level sequential wifi-based indoor localization. In *2021 IEEE/SICE International Symposium on System Integration (SII)*, pages 163–169, 2021.

[17] A. Kuzdeuov, M. Rubagotti, and H. A. Varol. Neural network augmented sensor fusion for pose estimation of tensegrity manipulators. *IEEE Sensors Journal*, 20(7):3655–3666, 2020.

[18] Binghao Li, Bruce Harvey, and Thomas Gallagher. Using barometers to determine the height for indoor positioning. In *International Conference on Indoor Positioning and Indoor Navigation*, pages 1–7, 2013.

[19] Jingbin Liu, Ruizhi Chen, Ling Pei, Robert Guinness, and Heidi Kuusniemi. A hybrid smartphone indoor positioning solution for mobile lbs. *Sensors*, 12(12):17208–17233, 2012.

[20] D. Lymberopoulos and J. Liu. The microsoft indoor localization competition: Experiences and lessons learned. *IEEE Signal Processing Magazine*, 34(5):"125–140", 2017.

[21] X. Dong T. Lu R. Westendorp M. T. Hoang, B. Yuen and K. Reddy. Recurrent neural networks for accurate rssi indoor localization. *IEEE Internet of Things Journal*, 6(6):"10639–10651", 2019.

[22] M. Nowicki and J. Wietrzykowski. Low-effort place recognition with wifi fingerprints using deep learning. page "575–584". International Conference Automation, 2017.

[23] F. Potortì, S. Park, A. Crivello, F. Palumbo, M. Girolami, P. Barsocchi, S. Lee, J. Torres-Sospedra, A. R. J. Ruiz, A. Pérez-Navarro, G. M. Mendoza-Silva, F. Seco, M. Ortiz, J. Perul, V. Renaudin, H. Kang, S. Park, J. H. Lee, C. G. Park, J. Ha, J. Han, C. Park, K. Kim, Y. Lee, S. Gye, K. Lee, E. Kim, J. S. Choi, Y. S. Choi, S. Talwar, S. Y. Cho, B. Ben-Moshe, A. Scherbakov, L. Antsfeld, E. Sansano-Sansano, B. Chidlovskii, N. Kronenwett, S. Prophet, Y. Landay, R. Marbel, L. Zheng, A. Peng, Z. Lin, B. Wu, C. Ma, S. Poslad, D. R. Selviah, W. Wu, Z. Ma, W. Zhang, D. Wei, H. Yuan, J. B. Jiang, S. Y. Huang, J. W. Liu, K. W. Su, J. S. Leu, K. Nishiguchi, W. Bousselham, H. Uchiyama, D. Thomas, A. Shimada, R. I. Taniguchi, V. C. Puschel, T. L. Poulsen, I. Ashraf, C. Lee, M. U. Ali, Y. Im, G. Kim, J. Eom, S. Hur, Y. Park, M. Opiela, A. Moreira, M. J. Nicolau, C. Pendão, I. Silva, F. Meneses, A. Costa, J. Trogh, D. Plets, Y. R. Chien, T. Y. Chang, S. H. Fang, and Y. Tsao. The ipin 2019 indoor localisation competition—description and results. *IEEE Access*, 8:206674–206718, 2020.

[24] C. Parini R. Bharadwaj and A. Alomainy. Experimental investigation of 3-d human body localization using wearable ultra-wideband antennas. *IEEE Transactions on Antennas and Propagation*, 63:"5035–5044", 2015.

[25] V. Renaudin, M. Ortiz, J. Perul, J. Torres-Sospedra, A. R. Jiménez, A. Pérez-Navarro, G. Martín Mendoza-Silva, F. Seco, Y. Landau, R. Marbel, B. Ben-Moshe, X. Zheng, F. Ye, J. Kuang, Y. Li, X. Niu, V. Landa, S. Hacohen, N. Shvalb, C. Lu, H. Uchiyama, D. Thomas, A. Shimada, R. I. Taniguchi, Z. Ding, F. Xu, N. Kronenwett, B. Vladimirov, S. Lee, E. Cho, S. Jun, C. Lee, S. Park, Y. Lee, J. Rew, C. Park, H. Jeong, J. Han, K. Lee, W. Zhang, X. Li, D. Wei, Y. Zhang, S. Y. Park, C. G. Park, S. Knauth, G. Pipelidis, N. Tsiamitros, T. Lungenstrass, J. P. Morales, J. Trogh, D. Plets, M. Opiela, S. H. Fang, Y. Tsao, Y. R. Chien, S. S. Yang, S. J. Ye, M. U. Ali, S. Hur, and Y. Park. Evaluating indoor positioning systems in a shopping mall: The lessons learned from the ipin 2018 competition. *IEEE Access*, 7:148594–148628, 2019.

[26] Q. Wan L. He X. Wang S. Bai, M. Yan and J. Li. Dl-rnn: An accurate indoor localization method via double rnns. *IEEE Sensors Journal*, 20(1):"286–295", 2019.

[27] Ahmed H. Salamah, Mohamed Tamazin, Maha A. Sharkas, and Mohamed Khedr. An enhanced wifi indoor localization system based on machine learning. In *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8, 2016.

[28] D. Tedaldi, A. Pretto, and E. Menegatti. A robust and easy to implement method for imu calibration without external equipments. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3042–3049, 2014.

[29] Weihan Tian, Diansheng Chen, Zihao Yang, and Hu Yin. The application of navigation technology for the medical assistive devices based on aruco recogni-

tion technology[*]. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2894–2899, 2020.

[30] Joaquín Torres-Sospedra, Raúl Montoliu, Adolfo Martínez-Usó, Joan P. Avariento, Tomás J. Arnau, Mauri Benedito-Bordonau, and Joaquín Huerta. Uji-indoorloc: A new multi-building and multi-floor database for wlan fingerprint-based indoor localization problems. In *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 261–270, 2014.

[31] Joaquín Torres-Sospedra, Raúl Montoliu, Sergio Trilles, Óscar Belmonte, and Joaquín Huerta. Comprehensive analysis of distance and similarity measures for wi-fi fingerprinting indoor positioning systems. *Expert Systems with Applications*, 42(23):9263–9278, 2015.

[32] Tsung-Han Tsai, Chih-Hao Chang, and Shih-Wei Chen. Vision based indoor positioning for intelligent buildings. In *2016 2nd International Conference on Intelligent Green Building and Smart Grid (IGBSG)*, pages 1–4, 2016.

[33] W. Zhang Y. Zhang W. Zhang, K. Liu and J. Gu. Deep neural networks for wireless localization in indoor and outdoor environments. *Neurocomputing*, 194:"279–287", 2016.

[34] Y. Wei and R. Zheng. Handling device heterogeneity in wi-fi based indoor positioning systems. In *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 556–561, 2020.

[35] F. Zhao Q. Wang D. Li X. Guo, W. Shao and H. Luo. Wimag: Multimode fusion localization system based on magnetic/wifi/pdr. page "1–8". International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2016.

[36] C. Xiang Q. Ye L. Liu Z. Wang X. He N. Yang X. Song, X. Fan and G. Fang. A novel convolutional neural network based indoor localization framework with wifi fingerprinting. *IEEE Access*, 7:"110698–110709", 2019.

[37] Zhengyi Xu, Jianming Wei, Jinxin Zhu, and Weijun Yang. A robust floor localization method using inertial and barometer measurements. In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8, 2017.

[38] G. Shen C. Zhao L. Li Y. Shu, C. Bo and F. Zhao. Magicol: Indoor localization using pervasive magnetic field and opportunistic wifi sensing. *IEEE Journal on Selected Areas in Communications*, 33:"1443–1457", 2015.

[39] Jun Yan, Lin Zhao, Jian Tang, Yuwei Chen, Ruizhi Chen, and Liang Chen. Hybrid kernel based machine learning using received signal strength measurements for indoor localization. *IEEE Transactions on Vehicular Technology*, 67(3):2824–2829, 2018.

[40] Guang Yang, Fang Gao, and Hao Zhang. An effective calibration method for wireless indoor positioning system with mixture gaussian distribution model. In *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, pages 1742–1746, 2016.

[41] H. Jiang Q. Zhu Y. C. Soh Z. Chen, H. Zou and L. Xie. Fusion of wifi, smartphone sensors and landmarks using the kalman filter for indoor localization. *Sensors,*, 15(1):"715–732", 2015.

[42] X. Li T. Yuan Y. Yang M. Wei Y. Zhang R. Sheng N. Grant C. Ling et al. Z. Zhong, Z. Tang. Xjtluindoorloc: A new fingerprinting database for indoor localization and trajectory estimation based on wi-fi rss and geomagnetic field. page "228–234". International Symposium on Computing and Networking Workshops (CANDARW), 2018.