

PRACTICAL AND ROBUST APPROACH FOR A NEURAL NETWORKS BASED INDOOR POSITIONING SYSTEM USING ULTRAWIDE BAND

By

Ryan Murari

Bachelor of Computer Engineering, Ryerson University, Ontario, Canada, June 2017

A thesis submitted to Ryerson University

In partial fulfillment of the requirements for the degree of

Master of Applied Science

in the program of

Electrical and Computer Engineering

Toronto, Ontario, Canada 2020

© Ryan Murari, 2020

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

ABSTRACT

Ryan Murari

Master of Applied Science

Electrical and Computer Engineering

Ryerson University, Toronto, Canada, 2020

With the increasing widespread of sensor technology, new solutions for indoor positioning systems are continuously being developed and with them, new services requiring accurate positioning data have seen a great rise in popularity. In this thesis, a new design technique and deployment methodology for an indoor positioning system using neural networks is proposed to offer more flexibility and simplicity in the development of such a system which is currently very context-bound. The usage of battery-powered tags implies also that systems should not require excessive power consumption and the large number of targets to position requires a method that is not only accurate but also scalable. The proposed positioning system utilizes a small “swarm” of neural networks tasked to position targets based on distance measurements from Ultrawide Band sensors and requires shorter fingerprint collection campaigns and enables more flexibility in system deployment and alterations. Instead of relying solely on real data collected on the field for the training of neural networks, synthetic data is used for an initial training phase. Together, these propositions allow flexibility in terms of adding, removing or altering positions of reference nodes and simplifies offline deployment operations of an indoor positioning system. This thesis presents a system operating in a laboratory-workshop environment capable of good positioning accuracies and maintains robust performances in poor signal propagation.

ACKNOWLEDGEMENTS

The author warmly thanks both his supervisors, Professor Muhammad Jaseemuddin and Professor Xavier Fernando for their offer to develop this system and for providing the support, tools and the Ryerson Communication Lab locals that lead to the work done in this thesis.

The author also thanks Peyman Moeini and Akbar Amari of Peytec Inc. that provided strong support and all of the material required for the development of the positioning system, as well as strong support and learning experience in the context of their operations, providing the insight needed take on the challenges related to this project. I also thank every graduate student I have met in the RCL locals who have provided me with great help and insight on how to tackle my thesis and helped in my self-teaching experience on neural networks-based systems.

My MASc program was funded in part by Graduate Fellowships endowed by Ryerson University and the Teacher Assistant jobs offered by Professor Reza Sedaghat whom I also graciously thank for the warm support and encouragement for my academic success.

Finally, I would also like to thank my family for all the love and support they have provided throughout the years.

Table of Contents:

Abstract	iii
List of Tables	vii
List of Figures	viii
Abbreviations	x
Chapter 1: Introduction	1
1.1 Overview and Motivations.....	1
1.2 Contributions.....	3
1.3 Objectives	5
Chapter 2: Indoor Positioning Technologies	6
2.1 Wireless Technologies	6
2.1.1 Wi-Fi	6
2.1.2 Bluetooth.....	7
2.1.3 Ultra-Wideband.....	8
2.2 Positioning Tools and Techniques	9
2.2.1 Tools used for Positioning	9
2.2.2 Core Positioning Principle	14
Chapter 3: Machine Learning Theory	17
3.1 Loss Function and Training	17
3.2 Neural Network Basics	18
3.3 Dealing with Multiple Answers: KNN Outlier Mitigation	20
3.4 Neural Networks and UWB in IPS	21
Chapter 4: System Model and Algorithms.....	23
4.1 Fundamentals of IPS design.....	23
4.2 System Model	24
4.2.1 Initialization and Offline Training Algorithm	24
4.2.2 Online Operation Phase	26
4.2.3 System Design	26
4.3 Early Results	29
Chapter 5: Experiment and Results.....	31

5.1 Experimental Procedure and Results	31
5.2 Evaluation and Analysis	45
Chapter 6: Conclusions and Future Works	46
References	47

List of Tables

Table 2.1: Overview of Strength and Weaknesses of Positioning Measurements	13
Table 2.2: Summary of Positioning Metrics Implementation Effectiveness	14
Table 3.1: Comparison of this thesis and relevant literature's works	21
Table 5.1: Average error of uncalibrated NNs.....	40
Table 5.2: Average error of calibrated NNs.....	44

List of Figures

Figure 1. 1: Decawave dwm-1001 development board	3
Figure 1. 2: Positioning System Workflow Diagram.....	4
Figure 2. 1: Ideal Example of Trilateration with 3 reference points.....	9
Figure 2. 2: Illustration of usage of KNN to identify outlier results.....	16
Figure 3. 1: Topology of a Neural Network	18
Figure 3. 2: Single Neuron.....	19
Figure 3. 3: Simplified model used in [31] and [33].....	21
Figure 3. 4: Simplified model of NNs used in this thesis' work	21
Figure 4. 1: Deployment Process Summary	25
Figure 4. 2: Typical Workflow of an IPS	27
Figure 4. 3: Positioning Process Structure	28
Figure 4. 4a: Cumulative Probability of error of NN vs trilateration in good condition	30
Figure 4. 5b: Cumulative Probability of error of NN vs trilateration in poor condition	30
Figure 4. 5: Set of early experimental results showing promising results for positioning with NNs	31
Figure 5. 1: High level view of positioning system structure	32
Figure 5. 2: Neural Networks creation and initial training logic	34
Figure 5. 3: Calibration Training Process	35
Figure 5. 4: Ryerson Communication Lab.....	36
Figure 5. 5: Dataset 1	37
Figure 5. 6: Dataset 2	37
Figure 5. 7: Dataset 3	38
Figure 5. 8: Reader 1 at coordinates (100cm, 200cm).....	38
Figure 5. 9: Stand holding 2 tags 1 meter above the ground	38
Figure 5. 10: Ideal target output matrix with standard deviation of spread = 1.5.....	39
Figure 5. 11: Ideal target output matrix with standard deviation of spread = 0.5.....	39

Figure 5. 12: Convergence of cost in during the initial training (NN-1 here)	40
Figure 5. 13: Example of True Output (left) vs NN Output (right)	40
Figure 5. 14: Dataset 1 average error of NNs vs Trilateration.....	41
Figure 5. 15: Dataset 2 average error of NNs vs Trilateration.....	42
Figure 5. 16: Cumulative Probability of Error Distance on Dataset 1	43
Figure 5. 17: Cumulative Probability of Error Distance on Dataset 2 and 3	43
Figure 5. 18: Cumulative Probabilities of Error Distances on all datasets combined of KNN only applied to all NNs' answers (Blue), only to all Trilateration's answers (Red), and all answers combined (Green)	44
Figure 5. 19: Cumulative Probability of Error distance on Dataset 1 and 3 of Positioning System after calibration of the system using Dataset 2 and Synthetic Data	45
Figure 5. 20: KNN's cluster bias in Difficult vs Good measurement conditions (exaggerated example).....	46

Abbreviations

IPS	Indoor Positioning System
NN	Neural Network
DNN	Deep Neural Network
KNN	K-Nearest Neighbors
SVM	Support Vector Machine
ML	Maximum Likelihood
MSE	Mean Squared Error
TOA	Time of Arrival
TOF	Time of Flight
AOA	Angle of Arrival
TDOA	Time Difference of Arrival
RSS	Received Signal Strength
UWB	Ultrawide Band
CDF	Cumulative Distribution Function
BLE	Bluetooth Low Energy
R_{ijk}	Reader Group including Reader i, Reader j and Reader k
NN_i	Neural Network assigned to reader group i
MPP	Main Positioning Process
MLP	Multilayer Perceptron

Chapter 1: Introduction

Localization systems, for positioning or tracking purposes, do not lack research upon, but are still in dire need for more practical and adaptable solutions. Current popular options for Indoor Positioning Systems (IPS) [1,2], tend to have either low accuracy [7], require extensive pre-deployment preparations such as long fingerprint collection campaigns [1], or simply rely on expensive hardware installations by adding as many nodes as needed to meet the requirements. As new wireless technologies continue to be introduced or made more accessible, different solutions are continuously developed to match the requirements of the given installation which are often only applicable to specific contexts. The research in this thesis aims at exploring IPS solutions exploiting Neural Networks (NNs) to develop a practical and accurate method, as well as scalable and robust against environmental interferences and sensor imprecisions; all while remaining relatively simple to deploy and cost-effective. It is also important to note that the mobile devices to localize are often limited in processing power as well as battery capacity which is also an important factor to address when developing an IPS.

1.1 Overview and Motivations

Indoor positioning technologies provide desirable benefits to consumers, whether for entertainment, convenience or utility. Some common examples of applications include object tracking, inventory management and augmented reality. While technologies such as GPS can reliably meet most positioning demands outdoor [8,9,10], and are now extremely widespread, they are incapable to do so in indoor environments.

In the past, and still today, the prime choices for most indoor positioning systems include, but are not limited to, Wi-Fi, ZigBee and Bluetooth Low Energy (BLE) due to their simplicity and availability. However, none of these technologies is made to provide precise ranging measurements and force the IPSs based on it to rely on creating large fingerprint database to match incoming positioning requests, or received signal strength (RSS) to distance conversions in order to locate a target. On the other hand, the most accurate beacon based IPS solutions are the ones using ultrasounds [11,12,13] or UWB radio signals [14,15,16]. However, ultrasounds have the significant drawback of very limited maximum range and may not penetrate obstacles and have

rather high-power consumption. UWB based positioning systems have potentially higher maximum accuracy than Wi-Fi or BLE (~20 cm in LOS [14]) and displays significantly larger coverage area per beacon installed (theoretically over 100 m in LOS). It also has the ability to penetrate obstacles and avoid most multipath interferences thanks to its larger bandwidth spectrum, qualities that justify its rise in popularity for positioning systems. However, as explained in [15], it remains a challenging task to develop a cheap and accurate IPS with UWB. Its qualities are significantly degraded in indoor environments, where LOS measurements cannot be guaranteed and are difficult to identify. Additionally, the costly hardware makes it difficult to justify the addition of too many nodes. For example, [18] shows that the maximum coverage area of a beacon is reduced to less than 15 meters in apartment type environments.

UWB distinguishes itself from other technologies thanks to its ability to measure “time of flight” (TOF or alternatively time of arrival, TOA) of transmissions and allows distance measurements to be made with far greater resolution and accuracy than signal strength-based ranging systems (20 cm as opposed to about 2 meters in the best cases for RSS based IPS). UWB research remains scarce partly as communications standards for this technology have only recently started to be established. More and more IPS delivering companies are emerging and making use of UWB and its low power, high accuracy capabilities; but their positioning algorithms mostly remain patented. Additionally, these systems require extensive preparation before deployment and may sometimes lack consistency over time due to changes in the environment.

There are now several UWB development kits available on the market for IPS solutions developers. Among them, the most common and pioneer of the field is Ubisense; but this thesis will center its focus around a newer alternative in the Decawave dwm-1001 development boards.

Mainly, this thesis studies the development and use of a precise UWB based IPS using a Deep Neural Network (DNN) based approach. Readings and experimental observations show that one weakness of UWB based IPS positioning process appears to be the rigidity of the trilateration (or other geometry-based solutions), leading to wide error margins even in mild measurement conditions such as partial alignment of the readers, signal reflections or large metal obstacles. Thus, the use of machine learning paradigm appears desirable. More specifically, using a NN to approximate the geometrical functions could result in a more robust positioning process while leaving the door open to the same enhancement methods typically applied to the geometrical

formulations such as KNN or Kalman Filters. The NN may find and take into consideration additional relationships between the readings on top of simply imitating the original formula. For example, automatically discarding results outside of the map boundaries.



Figure 1. 1: Decawave dwm-1001 development board

The research in both UWB positioning systems and NN based industrial positioning systems seems to be somewhat limited but advancements in such research could lead to a boom in indoor location-aware software and systems that could rely on a robust and accurate indoor positioning system.

Furthermore, it is hypothesized that the method described in this thesis could be generalized to be applicable in IPS using a different type and set of sensor data (RSS, Geomagnetic readings, etc.) but this option will not be covered in this document.

1.2 Contributions

Typically, an IPS cannot solely rely on the basic mathematical formulas to produce high quality positioning. Various optimization algorithms must be developed, usually to either process the input data before usage or to refine the output, enhancing it with averages, prediction algorithm or error mitigation schemes. It has been shown that the use of DNNs have consequential benefits for complex function approximations, resulting in more accurate output for far less complexity [6].

The neural network-based system developed and described in this thesis performs better than using the trilateration formulations in busy office/workshop conditions. To be precise, in interference prone conditions the system achieves an accuracy of 50 cm in poor propagation

conditions without requiring extensive training. Using UWB's capacity to provide good ranging measurements, the NNs can easily be trained with synthetic data for a specific map resolution, area and reader configuration. The number of inputs to the network models to generate can be increased if the data is expected to be available in real-time. Additionally, another contribution of this work resides in the fact that the system does not need multiple readings to find satisfying results, leading to potentially longer mobile device battery life.

Finally, the hardware requirements to run the networks are relatively low as all the experiments and simulations presented in this thesis were done using an Intel® Core™ i7-8550U Processor, a mobile CPU priced at about \$400 running in a VM environment. The training methodology of the NN could be compared to “fingerprinting” as the error-causing static interference, usually multipath, on the ranging measurements of the UWB readers tends to be similar for a given area. The NN also benefits from the same advantages of standard fingerprinting based IPS of providing accurate location estimation in the challenging multipath environments and thus can outperform some common localization techniques based on trilateration and

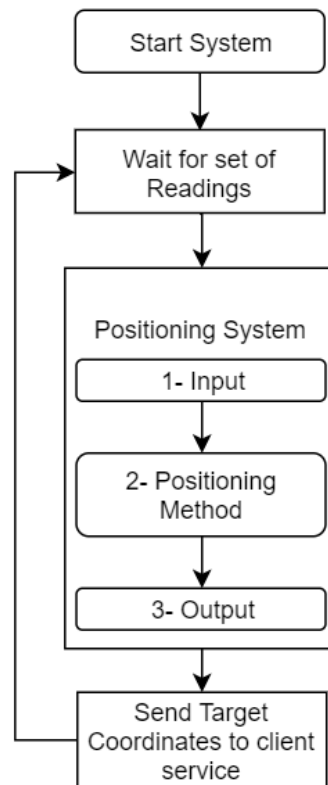


Figure 1. 2: Positioning System Workflow Diagram

triangulation. Furthermore, it distinguishes itself from standard fingerprinting methods by the fact that it does not require constant access to a database (the DNN weights can be thought of as the memory of the network's past training on the fingerprint database) and thus, can run locally. The positioning solution only requires the determination of the readers positions and, if existing, mobile devices of known location, the map dimensions and the grid resolution (20cm – 100cm is advised).

1.3 Objectives

In this thesis the focus lies on the development of an improved method of UWB-based indoor positioning system. To achieve this goal, it is necessary to accomplish the following individual objectives:

1. Show that emulated UWB readings can be used to train NNs to position targets in an area.
2. Obtain satisfying positioning accuracy without the aid of additional pre/post processing methods.
3. Present a calibration training algorithm using a low amount of fingerprint data to enhance the positioning accuracy of the system.
4. Create deployment methodology that is simple to reproduce and not overwhelmingly specific to each deployment locations.
5. Test the performances of the system in both optimal and interference prone environments to ensure robustness of the method.

Chapter 2: Indoor Positioning Technologies

The most fundamental and consequential aspect of an IPS is definitely the RF technology used and the algorithms employed alongside. In this part we explore the advantages and limitations of choices in technologies and algorithms for IPS.

2.1 Wireless Technologies

Most commonly used wireless communication technologies, such as Wi-Fi, Bluetooth and UWB, are primarily designed to communicate and provide data access. However, it is possible to translate some of the transmission signal's characteristics into distance measurements for the purposes of localization systems.

This section provides information regarding the various wireless technologies available and the ways they can be applied in indoor positioning systems. The technologies that will be discussed are Wi-Fi, Bluetooth and UWB which are not the only but certainly the most relevant in this study on indoor positioning systems.

2.1.1 Wi-Fi

Also known as the IEEE 802.11 standard is set to operate on the ISM band (Industrial, Scientific and Medical). Most of current mobile devices such as phones, laptops and various IoT devices are already Wi-Fi enabled. Its widespread makes documentation and cost of implementation very easy challenges to solve [20]. However, it is important to realize that the average Wi-Fi network is configured for maximized throughput to serve as a main data exchange mean in an area. Therefore, novel algorithms and protocols must be developed for the specific purpose of indoor localization. Additionally, it's large widespread makes its common frequency bands very busy and shown to create disturbances in the localization accuracy for some systems. Accurate Wi-Fi based IPSs will usually rely on RSS readings usually in fingerprint matching techniques as ToF measurements of Wi-Fi have errors in the order of 2 meters. IEEE 802.11 operates under different protocols with various frequencies: 2.4GHz, 3.6GHz, 5GHz and 60GHz;

all providing different range and speed of communication. Wi-Fi 2.4GHz has also the advantage of suffering less from NLOS conditions than typical higher band technologies [20]. However, the accuracy of Wi-Fi RSS-based positing systems can be seriously affected by attenuation of Wi-Fi signals by human bodies, dynamic obstacles and orientation of the mobile device. The effectiveness of Wi-Fi based fingerprinting system deployment is depends mainly from the time and efforts of site survey and maintenance of fingerprint database to keep it updated [21] which may not always be possible or worthwhile. Typical RSS based systems such as Horus [21] can only achieve about 2 meters accuracy and while more accurate systems have been developed [20, 23] and may achieve centimeter level accuracy. Often, accurate Wi-Fi based IPSs focus on the localization of smartphones and rely on their inherent high computational performances and quality which is not applicable for large scale inventory tracking for example. They remain impractical for industrial environment where the density of mobile devices to position may be very high and, amplifying battery life and delays concerns as Wi-Fi is not as energy efficient as Bluetooth or as fast as UWB.

2.1.2 Bluetooth

Also known as the IEEE 802.15.1 standard, Bluetooth is specifically defined by its particular MAC and PHY layers and their respective connectivity protocols. Bluetooth protocols have seen a significant increase in performances, energy efficiency and range in its latest upgrades; currently version 5.0 range is 40 up to 400 meters (LOS) as opposed to its previous versions ranging from 10 to 100 meters. The technology is primarily designed for communication over short distances and its original application was to act as a cable replacement protocol in wireless keyboards, wireless headsets etc. Its short distance low power characteristics makes Bluetooth very suitable in communication between small wireless devices. Bluetooth operates in the 2.4 GHz frequency band, part of the ISM. Interferences are minimized using a technique called Normal Frequency Hopping. This allows communications to change carrier among the several channels at high speed and in random sequence [27]. Bluetooth Low Energy (BLE) is the common power saving scheme adopted by indoor localization systems. BLE is capable of a throughput of 24Mbps. Bluetooth 5 provides, for BLE, options that can double the speed (2 Mbit/s burst) at the expense of range, or up to fourfold the range at the expense of data rate, and eightfold the data broadcasting capacity of transmissions, by increasing the packet lengths. In theory, localization techniques such

as RSSI, AoA, and ToF are applicable to BLE based systems, however, in practice, the typical BLE system relies mainly on RSSI based inputs to keep complexity of the system low [2, 5]; limiting the overall accuracy of the system. Two examples of commercial systems providing reasonable performances are Apple's iBeacons and Google Inc.'s Eddystone. However, these devices aim for a very particular goal and focus on proximity detection for specific applications [24]. Due to the short-range nature of Bluetooth, the amount of iBeacons units that need to be deployed for effective indoor use is significant. Its typical use involves predefined actions to take based on the relative proximity of the user to the beacons around him by consulting cloud-based servers.

2.1.3 Ultra-Wideband

UWB schemes use series of very short pulses with periods below 1 nanosecond over a bandwidth of over 500MHz. UWB is defined by frequencies ranging from 3.1GHz to 10.6GHz with very low duty cycles resulting in high power efficiency. Originally used for data transfer in a broad range of short-range indoor applications, it has become today a primary candidate for indoor localization as it is unaffected by the populated channel interferences from other typical wireless devices. It is also capable to penetrate obstacles more effectively than wireless technologies using smaller frequency band. It is also well shielded, but not immune, against the effects of multipath propagation, providing accurate ToF measurements thanks to its very short pulse duration. Due to the high time resolution of UWB signals, ToF based location estimation schemes usually provide better accuracy than the others. Systems like Ubisense are capable of accuracies up to 15cm [9, 14, 24]. This precise location estimation capability facilitates many applications such as medical monitoring, security, and asset tracking. Standardization efforts are underway in the IEEE 802.15.4a PAN standard, which will make use of the unique features of the UWB technology for location-aware sensor networking [25]. Further details on hybrid UWB positioning systems can be found in [25].

2.2 Positioning Tools and Techniques

2.2.1 Tools used for Positioning

A network consists of two types of nodes: anchors, or reference stations, are nodes with known positions, while agents or mobile devices (MDs) are nodes with unknown positions. This section presents various tools and techniques used along side wireless technologies to obtain positioning measurements.

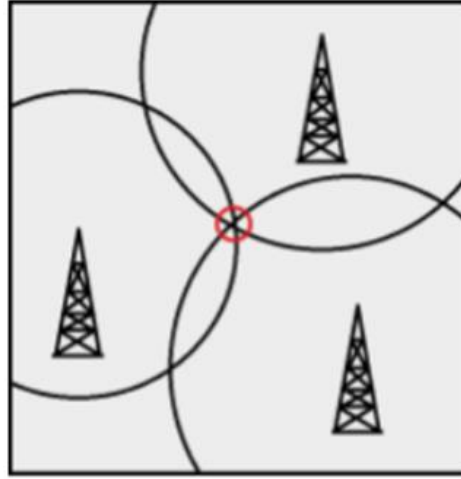


Figure 2. 1: Ideal Example of Trilateration with 3 reference points

A. Received Signal Strength (RSS):

The RSSI based approach for localization is known as the simplest and most widely used method for distance evaluation systems. It exploits the property of a signal power to fade over distance and thus, lower power at the received can be translated as indicator for longer distance from the transmitter. Typically, RSS is expressed in decibel-milliwatts (dBm). RSSI, on the other hand, is a relative measurement of the RSS and uses arbitrary unit scale predefined by the chipset vendor. For instance, Wi-Fi by Atheros generally uses an RSSI value between 0 and 60 and Cisco between 0 and 100.

An estimate of the distance “d” between transmitter Tx and Receiver Rx can be found from equation (1), combining RSSI value and a simple path-loss propagation model:

$$RSSI = -10n \log(d) + A \quad (1)$$

Where “n” is the path loss exponent and A the RSSI value at a reference distance from the receiver.

This technique has the advantage of being simple and very efficient in cost, however, it lacks overall accuracy. Signal attenuation sources such as walls or other obstacles or general RSS fluctuations caused by multipath fading phenomenon can, and will, lead to high inaccuracies in the distance measurements. While various methods can help reduce the error margin, such as using additional readings, only more complex algorithm (that may be combined with RSS based approach) can be expected to yield high precision positioning.

B. Channel State Information (CSI):

While the RSS remains more popular for its simplicity and low hardware requirements, Channel Impulse Response (or Channel Frequency Response), is similar in principles to the RSS approach but deliver a higher level of granularity and precision to the measurements. CSI information captures both amplitude (like RSS) and frequency response to evaluate distance from the transmitter. In most cases, the CSI is expressed as a complex value:

$$H(f) = |H(f)|e^{j\theta(H(f))} \quad (2)$$

where, $|H(f)|$ is the received amplitude and $\theta(H(f))$ the phase of the channel frequency. However, even with its better accuracy, this method is worth using if a network interface controller was already built to transmit such data and may result in a more costly solution than simply using RSS.

C. Fingerprinting:

This method requires a prior contextual analysis or environment survey, in which the various measurable relevant features such as RSS and/or CSI typically (but other relevant features can sometimes be used) readings are initially collected in the area to monitor. Upon deployment of the localization system, those measurements are compared with the measurements received from the target device to determine an estimate of the location (usually within a grid or known set of points).

Ultimately, fingerprinting combines all the available positioning tools to provide the best guess. When comparing the offline measurements with the current online ones, various algorithms can be used and combined to achieve higher accuracy:

i- Probabilistic approach: a machine learning matching model is built to determine the likelihood of the current data to match the data of all positions and the most likely match is selected as the location. There are various ways to process the data in order to obtain more accurate results. Furthermore, advanced probabilistic systems may also be able to determine the position beyond the predefined points (for example by averaging the position of all nodes times their likelihood). In theory, the potential precision of such system can be improved simply by increasing the number of measurements done offline prior to deployment. However, this method is weak against drastic changes in the environment if the offline measurements aren't updated. This approach is weak to time variant contexts.

ii- Artificial Neural Networks (ANN): this approach is comparable to the probabilistic method in implementation and behavior. Initial measurements of one or a set of features is done over the monitored zone. The data set is then used to train the ANN which can then be used to determine the location based on the online measurements of the selected features (and the other features if possible). [31, 32, 33]

iii- k-Nearest Neighbor (kNN): The k-nearest neighbor algorithm relies upon other principles to determine a set of likely positions of the target and, by selectively discarding the outlying results, selects k-nearest positioning matches which are then averaged to obtain the location. The challenge becomes the determination of the nearest matches, one way relies on offline measurements to guess which readings are the most reliable [28].

iv- Support Vector Machine (SVM): SVM are typically the method of choice for classification (or even regression) when it comes to machine learning for localization systems [28]. This scheme is comparable to the probabilistic approach and may also be used to implement any system using prior survey of the environment. The challenge in this case is the careful selection of the training parameters and coefficients. Typically, SVM are much less computationally demanding, which makes it an attractive solution for portable devices, with low energetic resources [29, 30].

D. Angle of Arrival (AoA):

AoA based approaches require an array of receiving antennas to determine which direction the received signal came from. The advantage of this method is that as low as two receivers can be used to find the position of a target on a plane. The biggest drawback of this scheme is that the high accuracy is very rapidly lost as distance increases where small angle errors translate in larger distance errors; and the hardware required is much more complex and expensive. Additionally, the AoA approach is very susceptible to multipath effects, making it a poor candidate for efficient indoor localization systems.

E. Time of Flight (ToF):

ToF exploits the physical property that is the speed an electromagnetic signal, the speed of light: $c \approx 3 * 10^8 \text{ m/sec}$. The distance between reader and receiver is computed by multiplying c with the signal propagation time from the transmitter. Ultimately, the computed distances can be used to triangulate the position information.

This technique requires rigorous synchronization between transmitter and receivers, usually basic handshake protocols can take care of synchronization and timestamp exchanges such as in UWB sensors. Measurements are highly dependent on the type of signal technology used, higher sampling rate results in higher accuracy. Large bandwidth is generally preferred.

F. Time Difference of Arrival (TDoA):

TDoA relies on the time difference between the arrivals of the signal at various synchronized receivers. For each receiver i , of known coordinates (X, Y, Z) , its distance to a transmitter j can be found with the following formula. To position an unknown target, a system of three equations (one per receiver) must be solved.

$$L_{D(i,j)} = \sqrt{(X_i - x)^2 + (Y_i - y)^2 + (Z_i - z)^2} - \sqrt{(X_j - x)^2 + (Y_j - y)^2 + (Z_j - z)^2} \quad (3)$$

Once again, once at least 3 different reference points have computed their distance from the device, triangulation can be used. Unlike ToF, this method benefits from only needing strict

synchronization between the fixed antennas and not the receiver. The accuracy is highly dependent on the bandwidth and the presence of line of sight [28].

G. Phase of Arrival (PoA):

This method uses the phase or phase difference between two reference stations of the carrier signal to measure distance from transmitter. The method assumes signals emitted are of pure sinusoidal. A first approach assumes the existence of a transit delay Di during flight and can be expressed as the quotient of the signal wavelength. The estimation in range is given by:

$$R_i = cDi/(2\pi f) \quad (4)$$

When PoA is applicable, TDoA will generally be a possibility as well. It is usually recommended to use in conjunction of ToF, RSS or TDoF to improve performances as it is much more susceptible to NLOS scenarios.

Metric used	Overall Hardware Cost	Synch. required	Min. # of measurements	Accuracy
RSS	low	No	3 (triangulation)	low
CSI	Medium (pre-built)	No	3 (triangulation)	medium
AoA	high	No	2 (line intersection)	High (lowers fast as distance increase)
ToF	low	Yes (Reference Station to Device)	3 (triangulation)	Depends on the signal frequency
TDoA	Medium - High	Yes (Among Reference Stations)	2	High in LOS
PoA	Medium	Yes (Among Reference Stations)	2	High in LOS

Table 2. 1: Overview of Strength and Weaknesses of Positioning Measurements

RSS	Easy and effective for many usages but lacks precision and may be unreliable in difficult environment
CSI	More reliable but will need discrete fingerprinting of the area
AoA	High accuracy with no fingerprinting but relies on directional antennae, complex algorithm and expensive hardware. Performances also drop quickly with distance
ToF	High accuracy with no fingerprinting but relies on synchronization of Tx - Rx, multiple reference stations with LOS as well as timestamps packets
TDoA	Accuracy with no fingerprinting and no synch needed with mobile device but synch needed among reference stations. Larger bandwidth and stamps usually needed.
PoA	Used with RSS, ToF and TDoF to boost overall accuracy, requires LOS for reliability

Table 2. 2: Summary of Positioning Metrics Implementation Effectiveness

2.2.2 Core Positioning Principle

Ultimately, the positioning processes used by an IPS can be divided into two main categories. The propagation approach, which uses the measured distance to multiple reference points, and the fingerprinting approach, which associates sets of measured feature characteristics with each location and tries to best match the incoming readings of the target.

In this section, some popular estimation techniques will be presented. Estimation techniques use the tools and metrics provided by the system and convert them to one or more possible coordinates to position the target.

A. Trilateration:

Trilateration is the most common and understood position estimation method. It relies on the system determining the distance of the target to at least three known reference points. From there, the equations of three circles with each Aps as their center can be written as:

$$(x - x_1)^2 + (y - y_1)^2 = d_1^2 \quad (5)$$

$$(x - x_2)^2 + (y - y_2)^2 = d_2^2 \quad (6)$$

$$(x - x_3)^2 + (y - y_3)^2 = d_3^2 \quad (7)$$

Where d_i is the measured distance from the target of coordinates (x, y) to a reference point i of coordinates (x_i, y_i) . The target's x and y coordinates can be found by solving this system of three equations:

$$X = \frac{\begin{vmatrix} (d_1^2 - d_2^2) - (x_1^2 - x_2^2) - (y_1^2 - y_2^2) & 2(y_2 - y_1) \\ (d_1^2 - d_3^2) - (x_1^2 - x_3^2) - (y_1^2 - y_3^2) & 2(y_3 - y_1) \end{vmatrix}}{\begin{vmatrix} 2(x_2 - x_1) & 2(y_2 - y_1) \\ 2(x_3 - x_1) & 2(y_3 - y_1) \end{vmatrix}}} \quad (8)$$

$$Y = \frac{\begin{vmatrix} 2(x_2 - x_1) & (d_1^2 - d_2^2) - (x_1^2 - x_2^2) - (y_1^2 - y_2^2) \\ 2(x_3 - x_1) & (d_1^2 - d_3^2) - (x_1^2 - x_3^2) - (y_1^2 - y_3^2) \end{vmatrix}}{\begin{vmatrix} 2(x_2 - x_1) & 2(y_2 - y_1) \\ 2(x_3 - x_1) & 2(y_3 - y_1) \end{vmatrix}}} \quad (9)$$

The formulas above concern trilateration on a 2D plane, in order apply it to 3D spaces a fourth reference point measurement must be used and similarly, a system of four equations must be solved.

B. Proximity:

This method associates a target's position with its closest reference point. It is typically used in situations where the accurate location is not needed, and the targets are expected to follow very specific paths on which they must be located. This method is relatively cheap and simple to deploy, it will often be combined with cheap and short-range wireless technologies such as Wi-Fi or Bluetooth. Some examples of applications include museum interactive guides detecting which piece of the museum the user is looking at, or in mines where payloads are moved and must be tracked through different distant sites.

C. Fingerprint matching:

The idea of this approach is to let certain signal properties or sensor measurements identify a certain location. Different signal properties and their corresponding locations are stored in a

database effectively creating a map of the radio properties at different locations. The location of a mobile device is determined by matching detected signal properties with the reference points stored in the database. Besides the accuracy this approach also has the advantage of not relying on the placement of APs. In fact, the location of APs can be unknown, and a fingerprinting approach would still work.

Chapter 3: Machine Learning Theory and Usage

The work presented in this thesis relies mainly on neural network machine learning techniques. This part will present fundamentals of neural networks and learning algorithms.

Machine learning algorithms are typically categorized between the following three classes:

→ Supervised Learning: this is the most popular scheme. It attempts to teach a learning function to map an, or set of, input to an output based on examples of input-output pairs. The end goal is to realize a function that can predict the output of unseen examples through generalization. This training method typically requires large amount of sample data.

→ Unsupervised Learning: in this case, the sample data output class is unknown, and the learning function is tasked to cluster data in different classes without prior knowledge of the output.

→ Reinforcement Learning: this class aims at continuously training the learning function through trial and error. Given some learning rules it can take decisions based on the current input-output pair.

The system presented in this thesis essentially relies on supervised training and mild learning rules that are comparable to reinforcement learning paradigm.

3.1 Loss Function and Training

Supervised learning algorithms progress by attempting to minimize a predefined cost function, also known as loss function. This function measures the gap between the goal output and the current predicted output. In its general form it can be expressed as:

$$J = y - \hat{y} \quad (10)$$

Where J , is the loss function, y the goal output and \hat{y} the predicted output value. It can be computed by running the given learning function, such as linear regression or a neural network with the related input vector. An optimization problem seeks to minimize the cost or loss associated

with sample input-output pairs. Details on the utilized loss function for this thesis' work are given in later parts.

3.2 Neural Network Basics

The concept of Neural Networks (NNs) is rooted in the attempt by scientists and mathematicians to mimic the way biological neuron networks learn and evolve. While standard systems and algorithms are implemented with a clear design and logic in mind before hand, NNs can “learn” to identify relationships and logic when observing large amounts of sample data. Fundamentally, to train them, NNs only require sets of input and desired associated output (supervised learning) as training data to find logical relationships. Although the core concept is simple to understand as it is very relatable to the way humans learn, it remains extremely difficult to predict the performances and their evolution of neural network as even slight changes in its topology, training methodology and most learning parameters, can result in great differences in performances and/or training required.

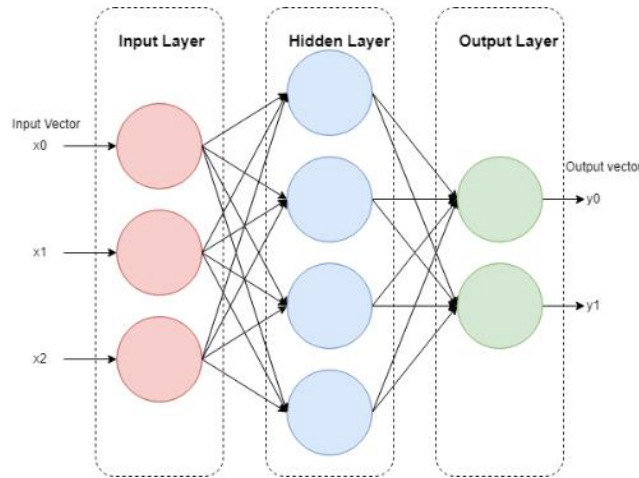


Figure 3. 1: Topology of a Neural Network with 2 input neurons 1 hidden layer of 4 neurons and 2 output neurons. All neurons from one layer are connected to all neurons in the next, making the network Fully Connected

In short, an NN is formed by a collection of neuron units arranged and connected in particular ways (typically in layers). The task of a neuron is to take the inputs of all the connected neurons in lower layers of the network, process it based on a mathematical function called

“Activation Function”, and output its result to the next connected neurons for additional process or output. The process of feeding an input and computing the output of a neural network is called Feed Forward Propagation and the process of correcting the connections between the neurons by computing its error margin, to “teach” the NN, is called Backward Propagation. The most popular types of NNs used today include, but are not limited to, Deep Neural Networks (DNN), Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), which are all derivative from the original Artificial Neural Network (ANN) composed of three layers of neurons, the input layer, one hidden layer and output layer, where each neuron in a layer is connected to all neurons in the next layer, it is said to be “fully connected”.

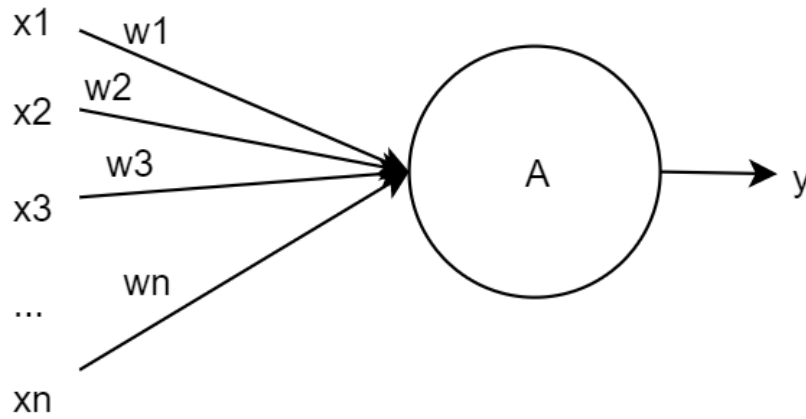


Figure 3. 2: Single Neuron

The figure above illustrates the operations done by a single neuron. The input $x_1, x_2, x_3, \dots, x_n$ are multiplied by the corresponding connection weights $w_1, w_2, w_3, \dots, w_n$ then summed and fed into the activation function A of the neuron, which outputs its result, y .

$$y = A\left(\sum_{i=1}^n x_i w_i\right) \quad (11)$$

The neural networks used in this thesis’ work belong to the category of Deep Neural Networks (DNNs) as they have 2 hidden layers which gives them the ability to find solutions to more complex non-linear problems. They are said to be feedforward networks as their actual usage, beyond training, solely relies on feedforward propagation. Most DNNs fit in the Multilayer

Perceptron category of NNs as they rely on backward propagation in supervised learning environments to be trained and use nonlinear activation functions.

3.3 Dealing with Multiple answers: KNN Outlier Mitigation

The K-Nearest Neighbour (KNN) algorithm is a method commonly used for classification and regression problems. The system presented in this thesis does not apply the KNN algorithm in a goal to sort between more than a single class (closest cluster to target). The algorithm is said to be a 1-nearest neighbour algorithm which is very intuitive and low in complexity. The quality of the results, given a mostly plausible set of answers, increases as the answer set size increases.

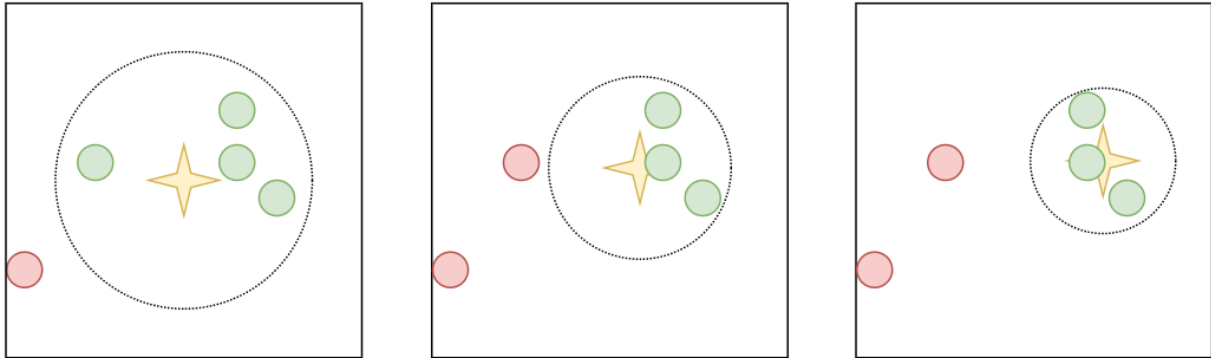


Figure 2. 2: Illustration of usage of KNN to identify outlier results

As illustrated in the figure above, once a set of plausible positions is generated, the average position is calculated. Sequentially, the farthest point from that average is removed from the result set and the average recomputed. This step is repeated until the farthest point's distance from the average is below a given threshold or that a minimum number of points is selected. An alternative option for the stopping policy may choose to stop the algorithm based on the average's movement after each iteration.

3.4 Neural Networks and UWB in IPS

The use and implementation of neural networks for indoor positioning has been explored by other researchers in the field and often differ in their approach to training data collection and usage. That is, works such as the ones presented in [31] and [33] use a single NN at the core of their positioning systems but [31] varies the amount and type of inputs to be fed into the NN while [33] investigates the accuracy variations with different dataset sizes used for training. On the other hand, [32] does not explore different training methods but rather aims at improving the quality of the training database content and adds UWB signal propagation parameters to the inputs of the NN to lead it into correcting error in measurements.

Positioning System	Fingerprint Database size (Locations x Readings/Locations)	Coverage Area (estimates)	Best Mean Accuracy (estimates)
[31]	302 x 4	20m x 4m	37cm
[32]	36 x NA (not mentioned) + 1155 x 11	2m x 4m x 1.5m	50cm
[33]	~ 108 x 10 (LOS) + 108 x 10 (NLOS)	70m x 4m	<50cm
This Thesis's	40 x 50 + 24 x 100 + 15 x 30	13m x 9m	33cm

Table 3. 1: Comparison of this thesis and relevant literature's works

However, each method often comes with its set of drawbacks such as in [33] in which slightly better performances are achieved but at the cost of an extensive fingerprint collection campaigns with measurements 1 meter apart from each other over a gallery about 36 by 4 meters. In [31] the inputs of the neural network are the TOA (ranging) measurements as inputs, similarly to this paper's system, and decided to add the RSS readings of the measurements, which ultimately slightly increased the performances of their system. The work in [31] also requires extensive fingerprint collection campaigns but ultimately achieves similar results or worse than this document's proposition in cases using 3 or less TOA measurements. The works of [31, 32, 33] all manage to achieve sub-meter accuracies using UWB.

Despite its capacity to make measurements at a decimeter precision, working with UWB also means that the system intends on deploying fewer UWB base stations and may not be able to make additional measurements as they are energetically costly to a mobile device. These factors reduce the maximum accuracy achievable. Therefore, it is still common to see RSS based systems

using Wi-Fi or BLE performing better than some UWB systems in some contexts, especially in complex environment as those systems often deploy a much larger amount of base stations, usually hopping to rely on existing infrastructure such as common Wi-Fi routers.

The works of in [31] and [33] chose to use data collected in the field for their training which provides high quality training data, if done properly. However, UWB technology can be exploited to achieve millimetre accuracies such as in the works of [34] but are then limited in coverage area and require more expensive setup as well as more power-hungry ranging protocols for the mobile device. One goal in mind when designing the proposed system was to lighten the data collection aspect of deployments and thus, elected to use synthetic data during training. Mainly, after their creation, the NNs are trained using generated data with some added noise until they can approximate trilateration (more on that in chapter 4 and 5).

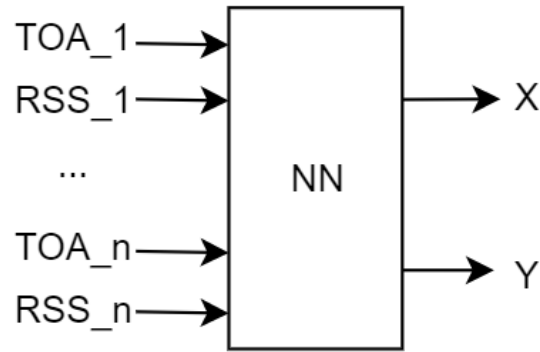


Figure 3. 3: Simplified model used in [31] and [33]

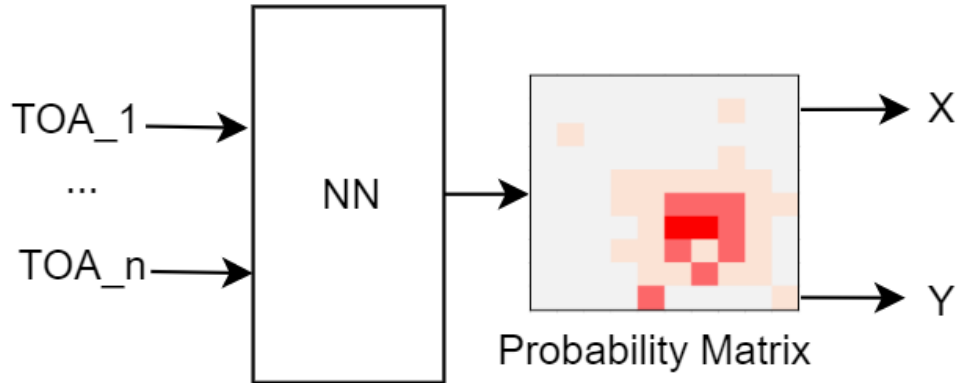


Figure 3. 4: Simplified model of the NNs used in this thesis' work

The approach to implementing NNs for IPS purposes appears usually to have a single NN trained on fingerprint data collected on the field and have it output the numerical coordinates of the target. While this leads to a more straightforward implementation of the IPS, such approach forces the use of larger NNs that are very specialized for a given deployment setup. A single change in the position of one of the reader nodes can render the NN unusable and force a complete retrain and a new fingerprint collection campaign which is very time consuming.

This thesis presents a method using a “swarm” of less complex NNs, assigned to 3 reader nodes each, allowing changes in the system without completely ruining the system’s performances. Additionally, the exclusion of less predictable fingerprint information such as RSS, allows the synthesis of “fake” training data which can be used to establish the first level of performance of the system. Furthermore, instead of directly outputting numerical coordinates, the NNs produce a probability matrix, allowing better visualization of the output and to introduce the concept of “closeness” to the training (more in chapter 5). Depending on the quality of the measurements, it is hypothesized that the matrix could be used to correlate outputs of the different NNs to provide better estimates; however, this approach will not be discussed in this document.

The approach presented in this document achieves relatively good mean error and did so using mostly synthetic data for training. Furthermore, another key aspect of the system presented resides in its ability to mitigate error coming from mismeasurements of one or more sensors. That is, systems presented in [31, 32, 33] use a single NN to process all the data which can be risky if some sensor suffers from momentaneous interferences and provides inaccurate readings. Given the usage of a “swarm” of NNs in the presented system, KNN can efficiently mitigate outlier results and possibly, minimize the effects of a mismeasurement. This advantage also benefits the system in its ability to save the power required from the mobile device to perform another measurement.

Chapter 4: System Model and Algorithms

The proposed solution utilizes Deep Neural Networks and KNN to aid trilateration in the positioning process. It results in more stable tracking of the target in poor signal propagation conditions and by extension, less measurements are needed to be taken which allows longer battery life on the mobile devices.

4.1 Fundamentals of IPS design

The process of designing an IPS can be initiated with the completion of at least the following few steps and decisions:

- **Requirements analysis:** Determine the required level of accuracy and feasibility. For example, in the case of tracking of payloads in a mine, the accuracy may be at the hundreds of meters level. Whereas, tracking various items in a supply chain could be satisfied with 1-meter accuracy.
- **System structure and topology:** Determine at which level is the computation done, what kind of network latency to expect and available processing power. Evaluate the possibility of introducing additional devices.
- **Positioning Principle:** As described in the previous chapters, the designer must choose the appropriate technique for positioning according to the needs and technology used.
- **Localization Process and Final Decisions:** Determine under what conditions must an input be discarded, and new measurements taken, and the kind of output that would be considered erroneous.

4.2 System Model

It is common for an IPS model to be separated into the offline or calibration phase, in which the system is prepared for deployment, hardware is installed and sample data is collected, and online phase, when the system is operational and describes its periodic tasks and activity routines and policies.

To clarify some terminology used in the rest of this document, some definitions are provided:

- **Reader:** reference station of known location performing the ranging measurements with the mobile devices. The location of a mobile device can be determined using a minimum of 3 ranging measurements with 3 different non-aligned readers.
- **Tag:** Mobile device of unknown location which initiates ranging measurements when set in motion. Tags have low computational capability and are battery powered, because it is expected to have a lot of them in circulation, low power consumption is critical to avoid extensive battery change operations.

4.2.1 Initialization and Offline Training Algorithm

Logically, the first step is to define the map of the area that will be monitored by the positioning system. The readers are installed on top of 2 meters tall pedestals and powered by the closest wall outlet. Additionally, a set of 40 tags is deployed at known locations and will be used as reference points (RPs) to collect fingerprint readings that will be used to further train the neural networks. The map is then represented as a grid of 50x50 cm cells* which will represent the output layer of the neural networks. As this project does not have a front end yet, the map dimensions and cell size must be written directly into the server.

**Cell size may be altered. However, it was observed that 50x50 cm cells were enough for our experiments and lower sizes would have led to NN dimensions too heavy to run on a personal computer.*

In the first phase, the system determines each possible combination of three and four readers to assign a neural network to, and proceeds to the initial training phase on synthetic data until convergence is achieved. Meanwhile, readings from the RPs are formatted for neural network training and saved.

$$\text{Number of NNs} = \binom{\text{number of readers}}{3} + \binom{\text{number of readers}}{4}$$

Given our test setup using 4 readers, we can train 4 NNs taking 3 inputs and 1 taking 4 inputs.

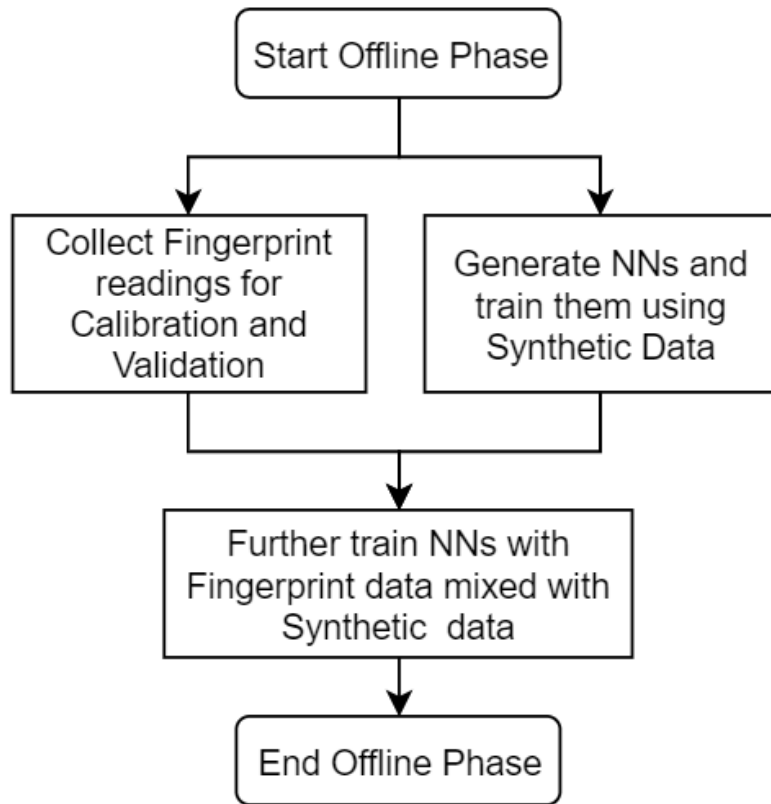


Figure 4. 1: Deployment Process Summary

In an actual deployment scenario, it is beneficial to be able to parallelize the fingerprint collection and initial training of the NNs to speed up the operations. Additionally, in a less complex environment and depending on the requirements, the NNs may provide acceptable accuracy even without the fingerprint training. Furthermore, assuming an already deployed system, adding a new

reader will not require retraining of all the NNs already in place and similarly, removing a reader will only result in the disabling of the NNs associated with it.

4.2.2 Online Operation Phase

Upon receiving a positioning request, the system runs the input through all valid NNs and computes all the possible trilateration's answers. Then, using the KNN algorithms, finds the final answer and uploads it to the cloud service.

The server running the system may also train the NNs periodically based on additional fingerprint data collected by fixed tags of known location. A performance comparison between the newly trained NNs and the current versions are performed using the offline fingerprint data and the real time data collected over time. If performance improvement is noticed, the new networks are individually selected to replace their current counterpart.

4.2.3 System Design

Based on the technology used, an IPS developer must determine the optimal positioning process available, or more often, combination of them. Improvements in performance are often found in enhancements of the input or output of the system. For example, better RSS to distance conversions can be achieved by combining Wasserstein distance interpolation with the sequential Monte Carlo method for tracking in [1]. Refining of the output can be done using Kalman filters for such as in [2] where the position of the tag device is estimated by adding the previous position and displacement from sensor-based positioning.

There are ultimately two main basic techniques used to position a target: trilateration and fingerprinting; from which other methods can also derivate. In trilateration, estimated distances are used to calculate the most probable coordinates using the geometry of triangles [3]. On the other hand, fingerprinting methods rely on correlating or matching the measurements with previously collected and mapped data, this can be done with machine learning algorithms or other general data correlation calculations. In this paper, we label the positioning technique of the IPS as the Main Positioning Process (MPP).

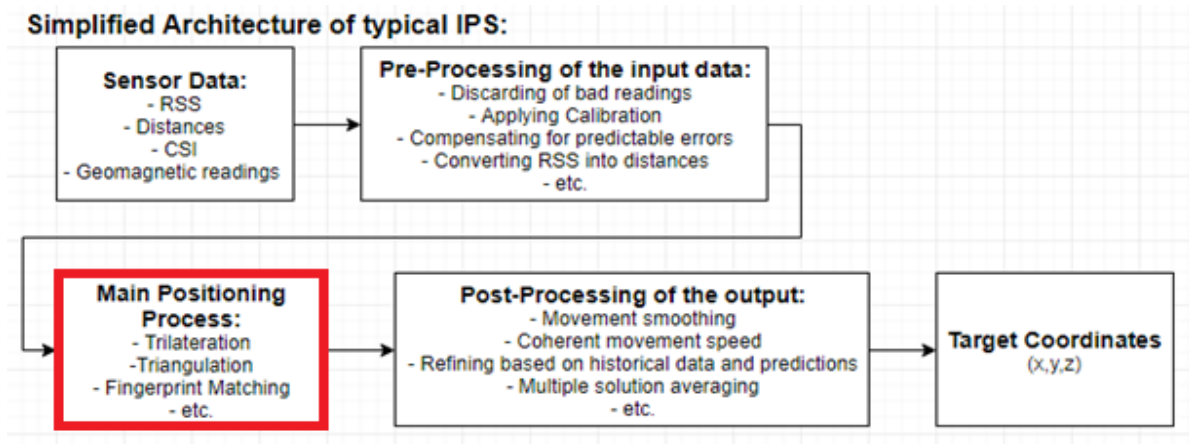


Figure 4. 2: Typical Workflow of an IPS

The proposition at hand targets UWB's most common MPP, trilateration. This process requires a minimum of three ranging measurements from three different non-aligned reference points. Therefore, given more than three such measurements, the process can be run as many times as unique groups of three readings can be made. Finally, the multiple possible answers are translated into a final set of coordinates, through weighted average or KNN algorithm.

This thesis' work introduces neural networks accomplishing the same task as trilateration, thus, providing a larger set of possible answers, increasing the likelihood that an algorithm such as KNN results in a more accurate answer. The structure of the proposed model is described in Figure 4.3. The number of different combinations of three inputs given "n" readings can be computed with the following formula:

$$\binom{n}{3} = \frac{n!}{3!(n-3)!}$$

As previously explained, the 1-nearest neighbor algorithm benefits greatly from a surplus of answers. The proposed method using NNs to generate additional answers then guaranties at least 2 answers in the worst case of only 3 readings and, given 4 readings, immediately raises the result set size to 9 (4 from trilateration, 4 from 3-Inputs NNs and 1 from 4-Inputs NN).

Furthermore, to maintain consistency in the movement tracking of a target, the previous position of the target is added to the original result set on which the 1-Nearest Neighbor (1NN) is applied. This scheme leads the algorithm to favor results closer to the previous target position and reduces the risk that a wrong answer clusters be selected.

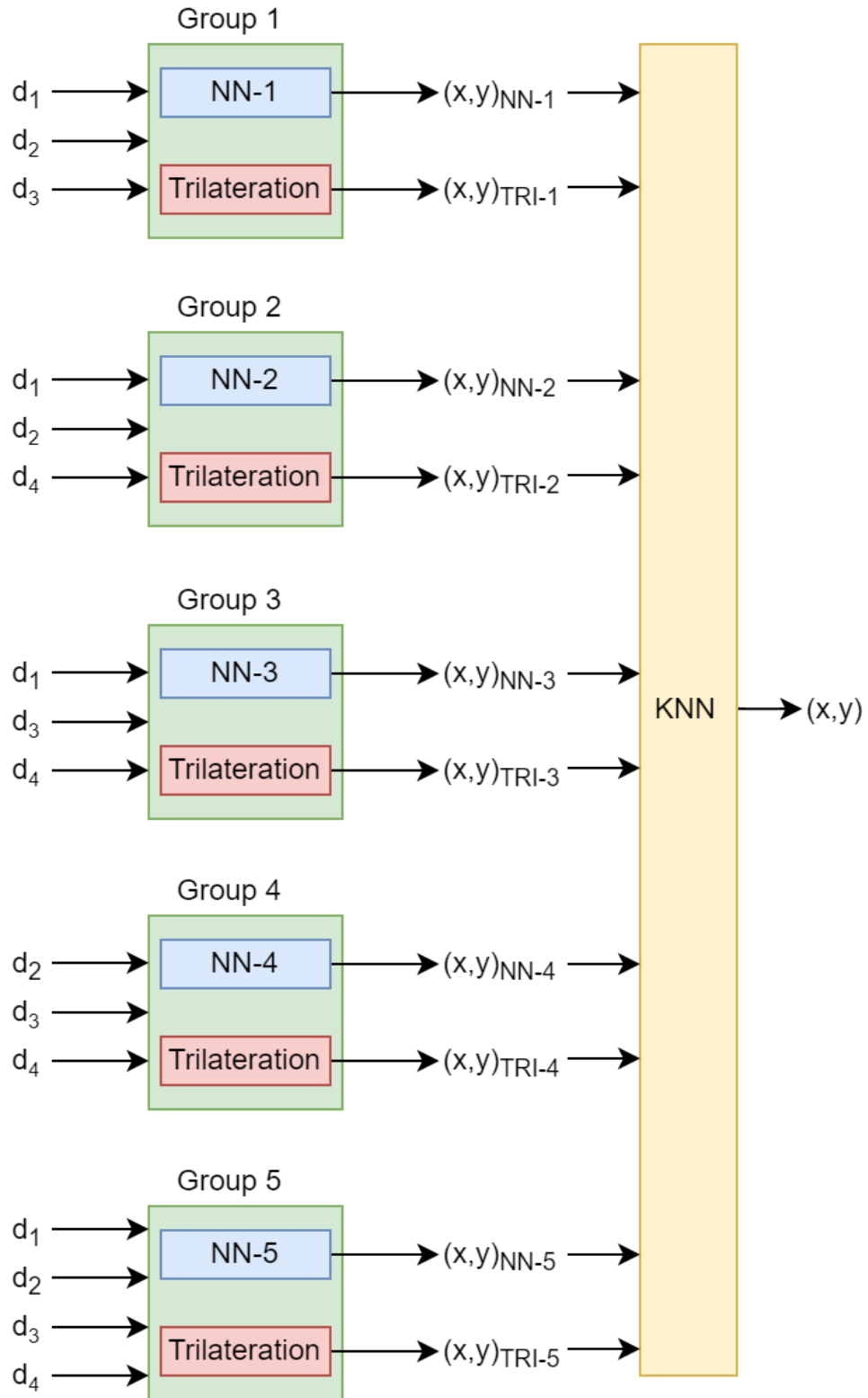


Figure 4. 3: Positioning Process Structure

4.3 Early Results

Figures 4.4 and 4.5 show results of performance comparisons between trilateration and equivalent NNs. The results clearly indicate higher accuracy given low quality measurements where the NNs achieve errors below 1.5m 90% of the time as opposed to 70% for trilateration.

Additionally, further test experiments were run to determine an appropriate depth and size for the NNs. It can be observed that DNNs with at least 200 neurons per layers and a depth of 2 or 4 perform noticeably better. The rest of the experiments in this document utilize NNs with 2 hidden layers and 200 neurons per layer in order to fit the available material specifications.

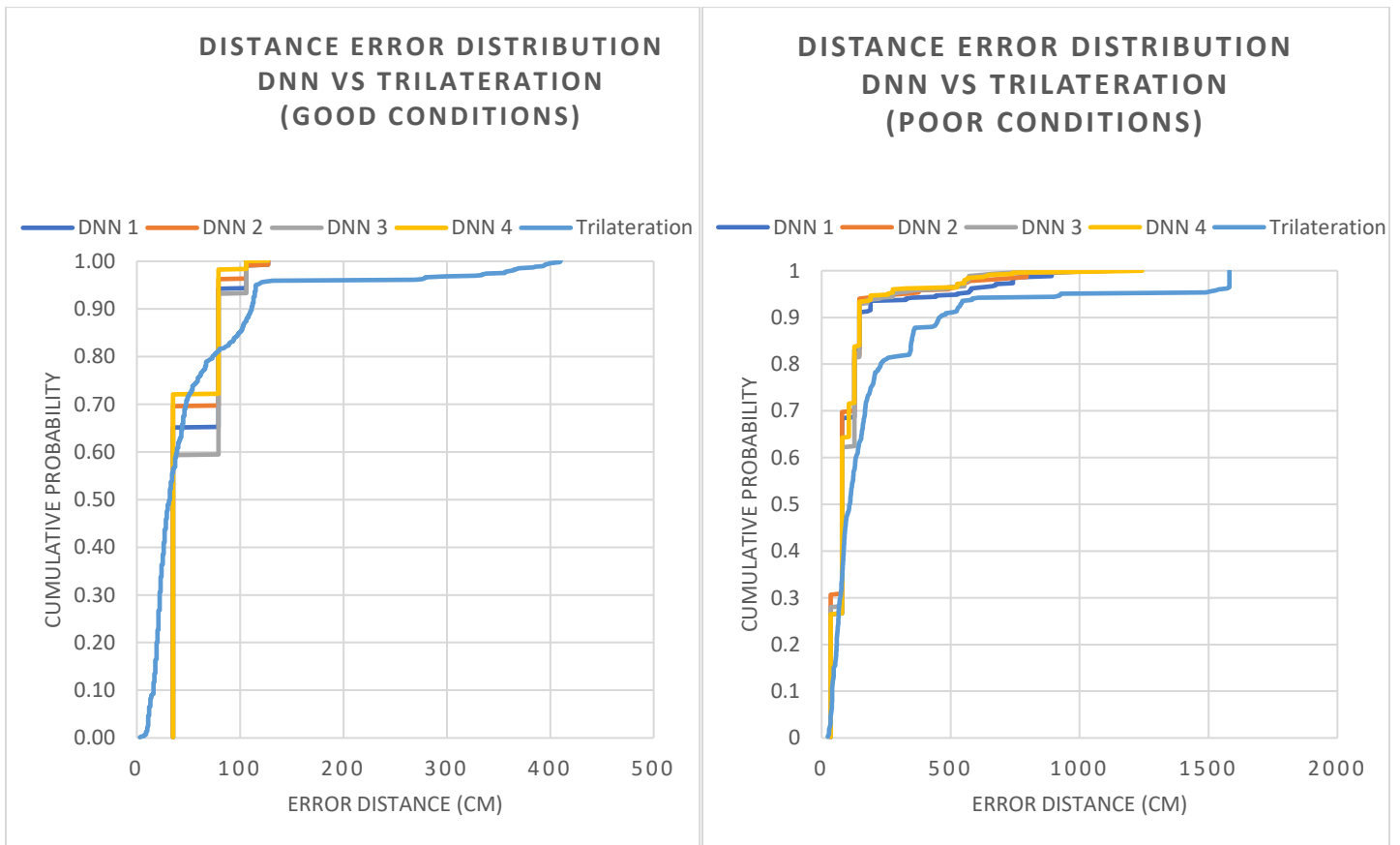
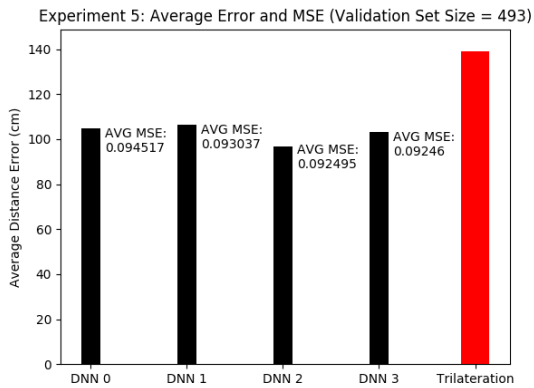
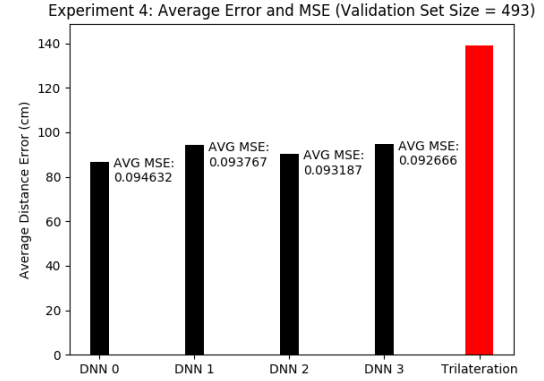
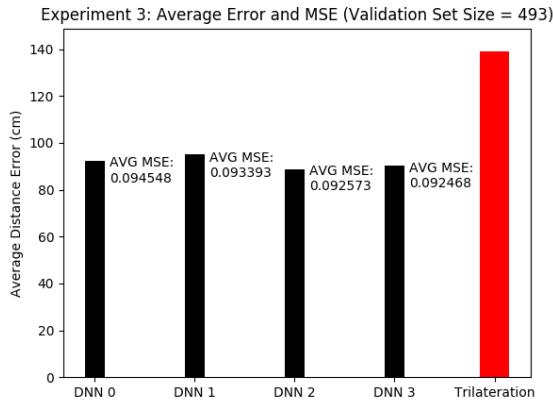
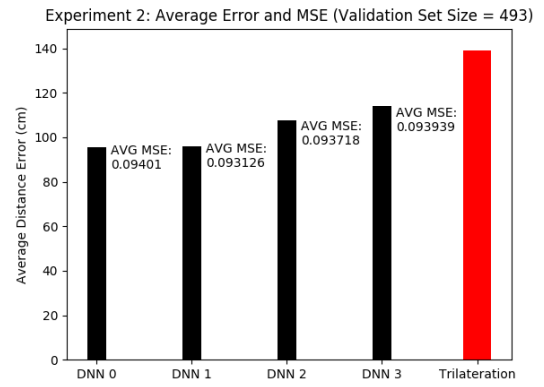
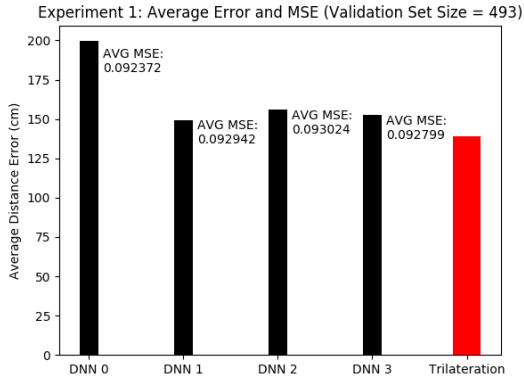


Figure 4. 4a: Cumulative Probability of error of NN vs trilateration in good condition.

Figure 4. 4b: Cumulative Probability of error of NN vs trilateration in poor condition.



Experiment 1, 2, 3, 4, 5 use respectively 1, 2, 3, 4, 5 hidden layers DNNs

DNN 0, 1, 2, 3 have respectively 100, 200, 300, 400 neurons in each hidden layer

Figure 4. 5: Set of early experimental results showing promising results for positioning with NNs

Chapter 5: System Experiments and Results

5.1: System Architecture

The positioning system's base is composed of tags, readers and a main server. Readers report to the server the ranging measurements they perform with each tag and the server oversees the running of the appropriate DNNs for a given set of data as well as the additional training sessions that may need to be performed.

The UWB modules used are the Decawave DWM1001 Development Boards with a specified 10 cm accuracy on ranging measurements. However, it is important to note that this metric highly fluctuates based on the actual distances and levels of interferences present during the measurement. The server role was taken by a personal laptop (Lenovo C930) equipped with an Intel Core™ i7-8550U CPU (1.8 GHz) and 16 Gb of RAM. Codes for the experiments were developed in a Linux Virtual Machine in python3 using Keras, an open source neural network library running on top of TensorFlow.

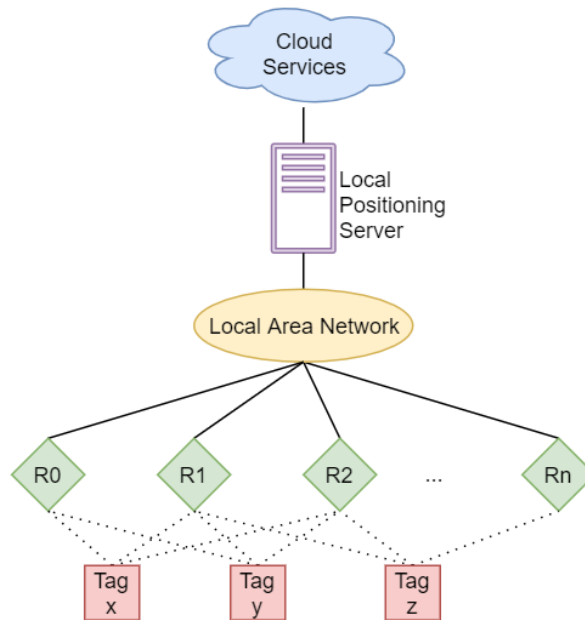


Figure 5. 1: High level view of positioning system structure

5.2: Neural Networks Creation and Calibration

Neural networks used in this thesis's work are initially trained using only synthetic data and then calibrated using fingerprint data collected prior to the experiments. Calibration was performed using alternating samples of synthetic data and fingerprints the database using large batch training. The synthetic data allowed the NNs to not be affected too significantly by very incoherent readings and not shift from its objective too significantly. Additionally, the large batch used for calibration training yielded slightly more stable learning and focus on overall performance improvements.

In the Linux distro, the following command can be used to install Keras and its requirements for python3:

```
>> sudo pip3 install Keras
```

Furthermore, the neural networks can be initialized with the following code where “numCell” is the total number of cells on the map:

```
in_sz = 3
out_sz = numCell
act = 'tanh'
out_act = 'sigmoid'
lossF = 'mse'
optim = 'rmsprop'
layers = [in_sz, 500, 500]
model = Sequential()
for i in range(1, len(layers)):
    model.add(Dense(layers[i], input_dim=layers[i-1],
        activation=act, init='uniform'))
model.add(Dropout(0.3))
model.add(Dense(out_sz, input_dim=in_sz, activation= out_act))
model.compile(optimizer=optim, loss=lossF)
```

Because the output of our network is a classification probability matrix, the sigmoid function was used for the output layer and tanh for the hidden layers, as recommended for complex function approximation in [6]. The choice was also supported by some trial and error tests, showing faster convergence and better output with these hyperparameters. Finally, while standard mean

squared error was chosen as the backend training loss function, the system evaluates loss every minute based on the average positioning accuracy of 200 fingerprint samples in which 100 are extracted from Dataset 1 and 100 are synthetically generated. Convergence was determined when no significant improvements in accuracy could be observed.

Once all the desired NNs are generated, they are saved in a directory to be used by the calibration algorithm and the positioning system:

- The calibration algorithm uses the datasets to further train the NNs and improve performances.
- The positioning system is tasked to use the saved NNs and possibly collect further training data with some fixed reference tags deployed on the field.

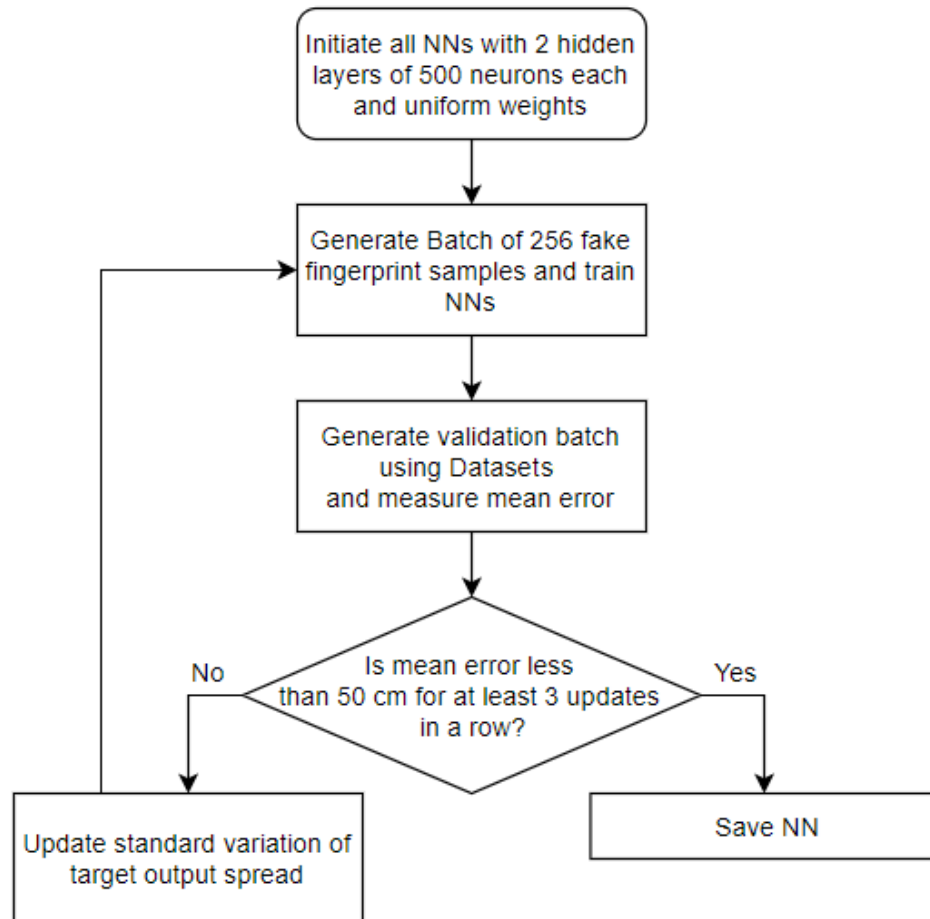
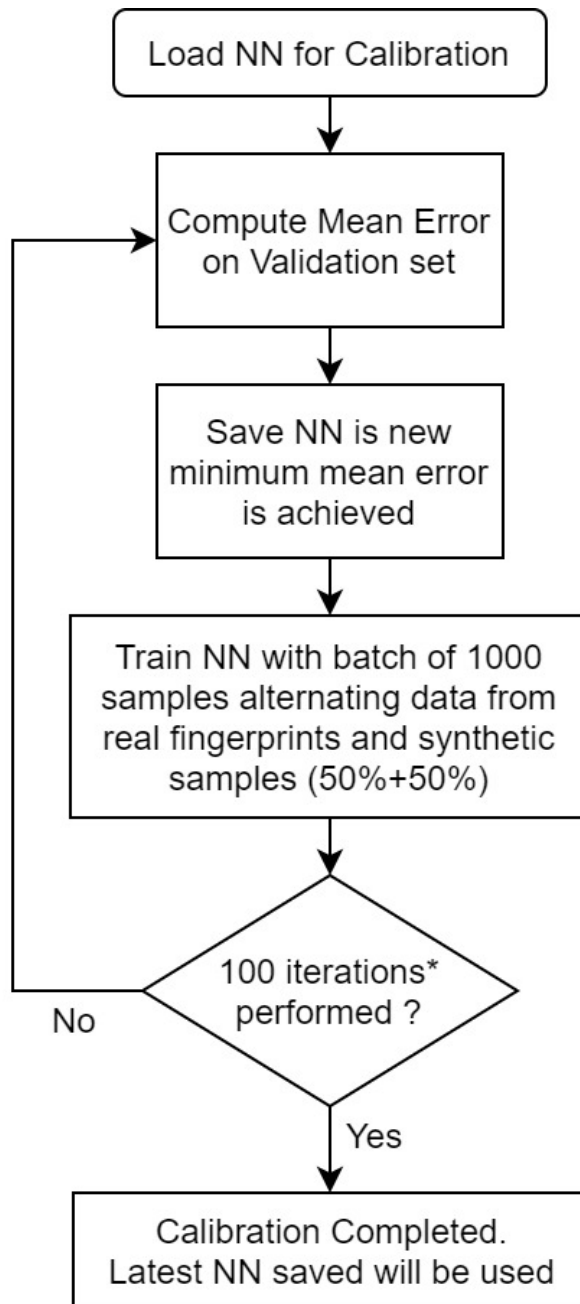


Figure 5. 2: Neural Networks creation and initial training logic



**The number of iterations is arbitrary, chosen during program testing based on processing speed and results observations.*

Figure 5. 3: Calibration Training Process

5.2 Experiment and Results

5.2.1 Experimental Procedure

The experiments were performed in the Ryerson Communication Lab (RCL – ENG401). This location may not be considered representative of an industrial environment, but it is nonetheless relevant as a test location for an IPS. The lab consists in 2 rows of wooden desks with metal legs and a total of 16 computers on top and 2 large metal desks holding various large electrical tools (oscilloscopes, printers, generators, 45” TV...). Below the desks is mostly filled with boxes of books, hardware and other. The side of the lab are ornated with 8 metal shelves and a fridge as well as a few other pieces of metal/wooden furniture.



Figure 5. 4: Ryerson Communication Lab

A. Fingerprint Database Creation

Three main fingerprint collection campaigns were performed in the lab in various conditions. Dataset 1's readings were taken in controlled conditions, with few to no people around and maintaining a constant height of 1 meter above the ground for the test tag placed on a stand; this dataset represents the scenario with better conditions (not ideal) relative to the other two. Dataset 2's measurements conditions were not as controlled as for Dataset 1: more people were in the office and little to no care for the height of the test tag, not placed on stand, typically between 50 and 150 cm above the ground. It was also noticed later that Reader 4 was misplaced by about 10 cm during that specific measurement campaign. This Dataset represents the scenario with typical conditions. During both previous procedures the readers were position on top of a stand 2 meters above the ground. Finally, Dataset 3 was built after having positioned the readers 50cm above the ground in order to provide more disturbance in the measurements and as there are various material boxes, metal bars and other tools filling the space under the desks.

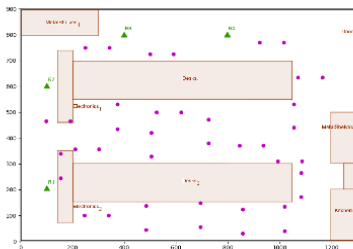


Figure 5. 5: Dataset 1

Number of Test Points: 40

Readings per point: 50

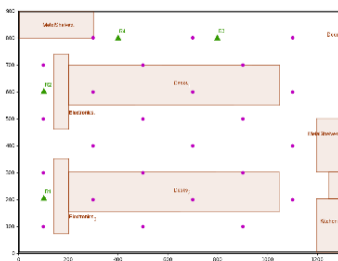
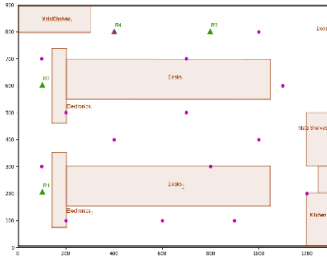


Figure 5. 6: Dataset 2

Number of Test Points: 24

Readings per point: 30



Number of Test Points: 15

Readings per point: 30

Figure 5. 7: Dataset 3



Figure 5. 8: Reader 1 at coordinates (100cm,200cm)



Figure 5. 9: Stand holding 2 tags 1 meter above the ground

B. Neural Networks Creation:

For the experiments, the map dimensions used were 1300cm x 900cm with a cell size of 50cm. The NNs were all trained for about 15 minutes each with synthetic data only using batches of 256 samples.

In the earlier stages of development of this project, it would have taken around 50 minutes per neural networks to train them with synthetic data; today, it takes about 5 minutes. This improvement was observed when weights were initialized as uniform rather than random and that tanh was used on the hidden layers rather than sigmoid.

The neural networks trained with batch method have their weights updated based on the cumulative backpropagation results of the batch rather than after each individual example. The training algorithm can be summarized in 3 steps:

- 1- Generate Batch of synthetic data with added random error of up to $\pm 25\text{cm}$ to each distance.
- 2- Train neural network (update weights)
- 3- Measure cost/performances on a validation set = average error over a large example set

The desired output is a probability matrix spiking at 1 in the matrix cell corresponding to the most likely location the target is in. This is best represented using a heat map, figure 5.9 and 5.10 represent the output matrix of a training sample where the target is located at (5m, 5m), in this case, using 25cm cells to provide better illustration.

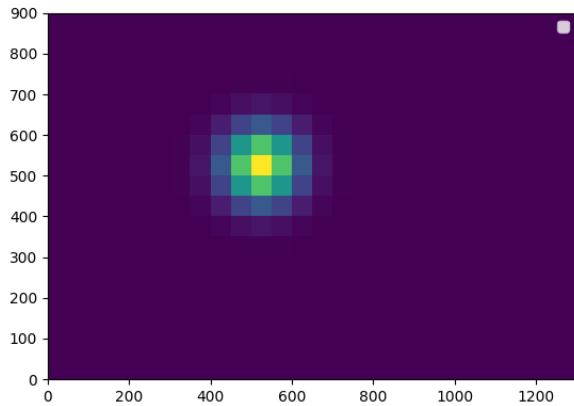


Figure 5. 10: Ideal target output matrix with standard deviation of spread = 1.5

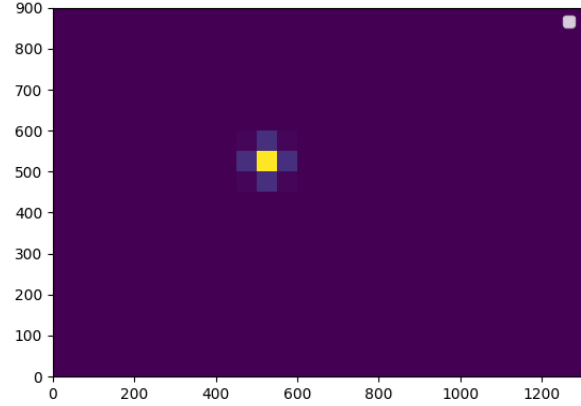


Figure 5. 11: Ideal target output matrix with standard deviation of spread = 0.5

To synthesize this output, a zero 2D matrix of dimensions $26 \times 18 = (1300/50) \times (900/50)$ is first created and the target cell value set to 1. Then, a gaussian filter is applied to create the spread around the target. The filter can be configured with different values of standard deviation that affects the amount of spread the final matrix will have around the target cell: higher deviation results in larger spread and vice-versa.

Originally, the synthetic data true output “target spread” used a static standard deviation of 1.25 (chosen semi-arbitrarily through trial and error). However, in later developments, that value was made to dynamically change during training with respect to the neural network’s performances:

the better the performances, the lower the allowed deviation would become. It was found that this change greatly speeds up the time it took for the synthetic training to yield acceptable performances.

The standard deviation for the target spread is now calculated such that:

$$\sigma = \max\left(\frac{\text{Current Average Distance Error (cm)}}{400}; 0.75\right)$$

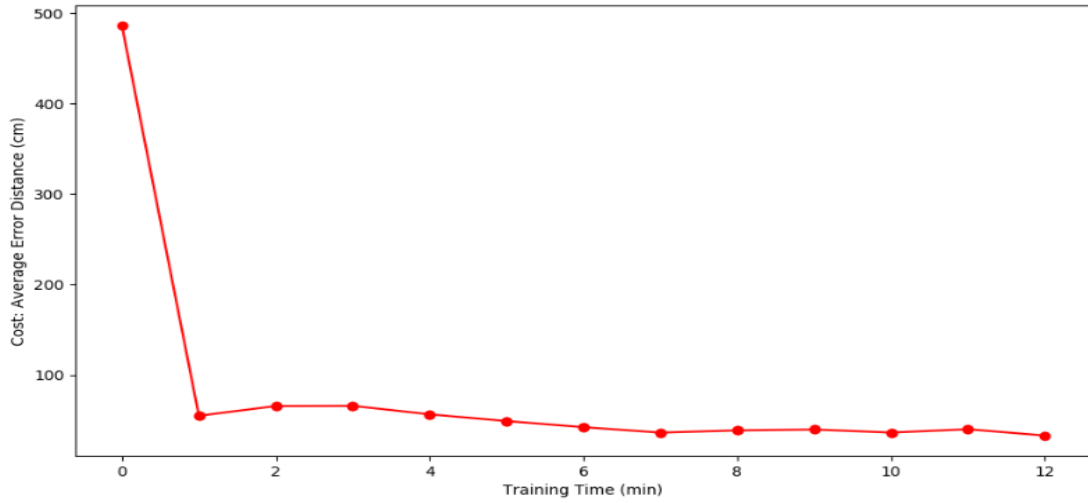


Figure 5. 12: Convergence of cost in during the initial training (NN-1 here)

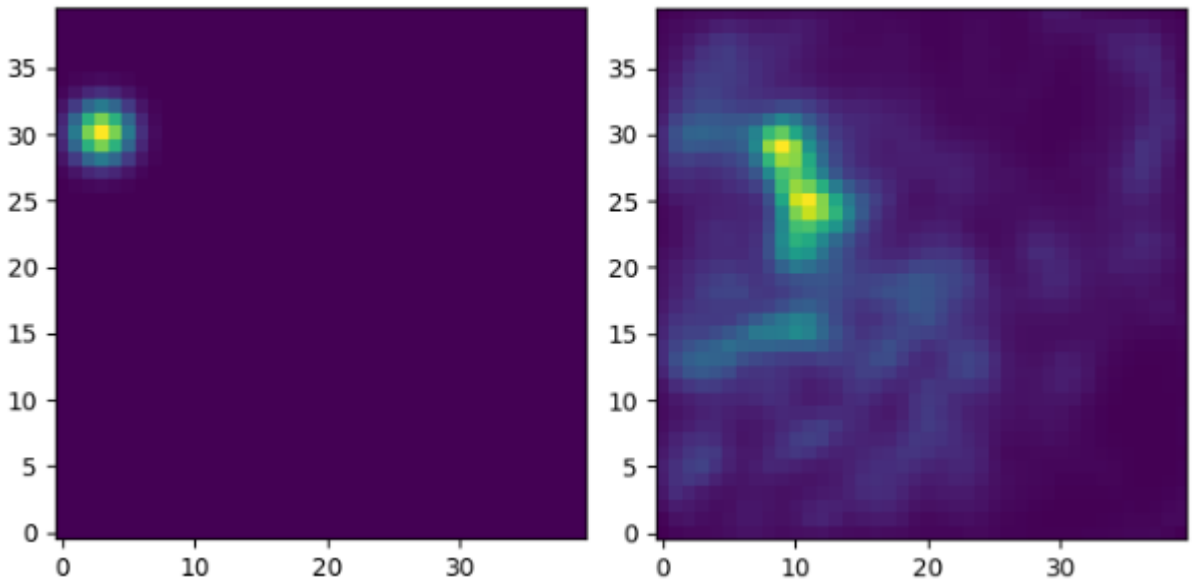


Figure 5. 13: Example of True Output (left) vs NN Output (right)

C. Performance Evaluation:

In the first set of tests, the average error of the uncalibrated NNs is computed on each Datasets.

<i>NN Group</i>	<i>Mean Error (cm)</i>		
	Dataset 1	Dataset 2	Dataset 3
<i>R1-R2-R3</i>	66	43	68
<i>R1-R2-R4</i>	66	62	92
<i>R1-R3-R4</i>	86	59	77
<i>R2-R3-R4</i>	54	70	88
<i>R1-R2-R3-R4</i>	57	44	72

Table 5. 1: Average error of uncalibrated NNs on entire Database

From Table 5.1, it can be said that the performance of uncalibrated NNs is not satisfying for system requiring high degrees of accuracy. However, the NNs results can be said to be at least promising when put against their respective trilateration results (see Figure 5.14, 5.15). The larger than normal error of trilateration for readers 2, 3 and 4 can also be explained by the fact that it was the “most aligned” set of reader.

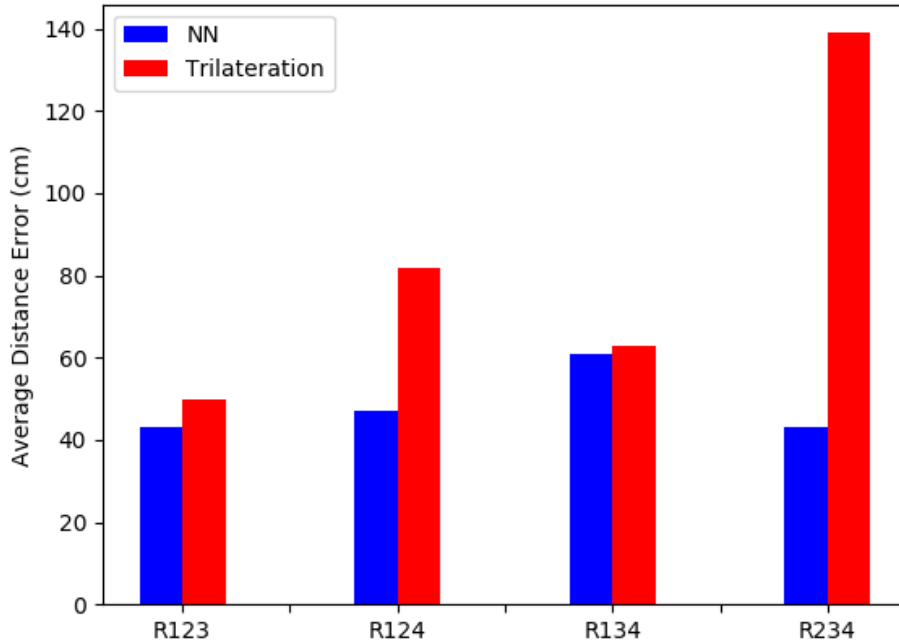


Figure 5. 14: Dataset 1 average error of uncalibrated NNs vs Trilateration

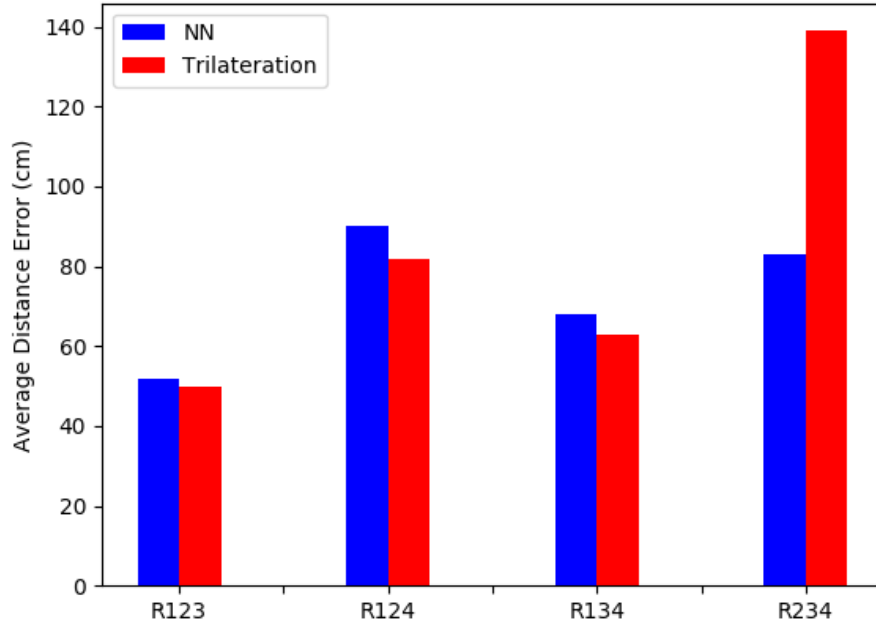


Figure 5. 15: Dataset 2 and 3 average error of uncalibrated NNs vs Trilateration

Figure 5.15 can clearly demonstrate the lower quality of Dataset 2 and 3 with respect to the first. However, rather than simply looking at mean error, when analyzing an IPS' performances the rate of distance error is a metric that is as important as the mean error (can relate to deviation).

In the following pages, the rates of distance error of the NNs is analyzed in order to identify the specific strengths that can be extracted from the proposed method.

When the rate of error distance of the NNs was compared to trilateration (Figures 5.17, 5.18), a conclusion was made: the NNs appear to be more robust to measurement errors but provided limited maximum accuracy as opposed to trilateration, which was capable of far more accurate positioning when the readings were of higher quality. In Figure 5.17, it can be observed that the NNs limited the error to a smaller minimum than trilateration for each positioning groups, but the best answers were provided by trilateration. Further analysis at the performances on Dataset 1 reveals that in cases where the error distance is lower than ~40cm, trilateration tends to provide better results more often. However, the neural networks errors are all bellow 140cm while trilateration's error can go as high as 200cm.

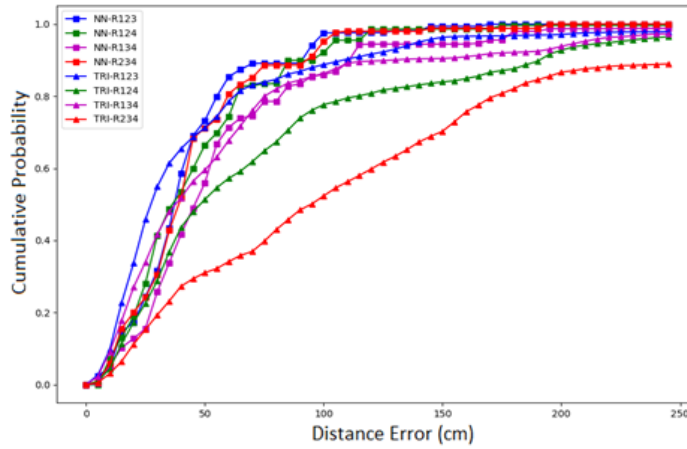


Figure 5. 16: Cumulative Probability of Error Distance on Dataset 1

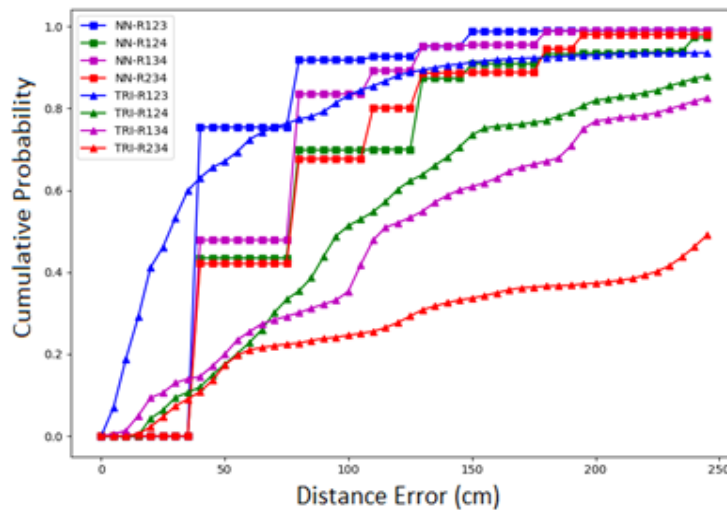


Figure 5. 17: Cumulative Probability of Error Distance on Dataset 4

In order to combine the benefits of trilateration's low error when the quality of readings is high but maintain the robustness of the neural networks with less optimal measurements, KNN outlier mitigation algorithm was used to sort through all the results of both neural networks and trilateration. The performance of the new system using KNN to sort through all the answers was compared to KNN using only the trilateration results and only the neural networks results respectively.

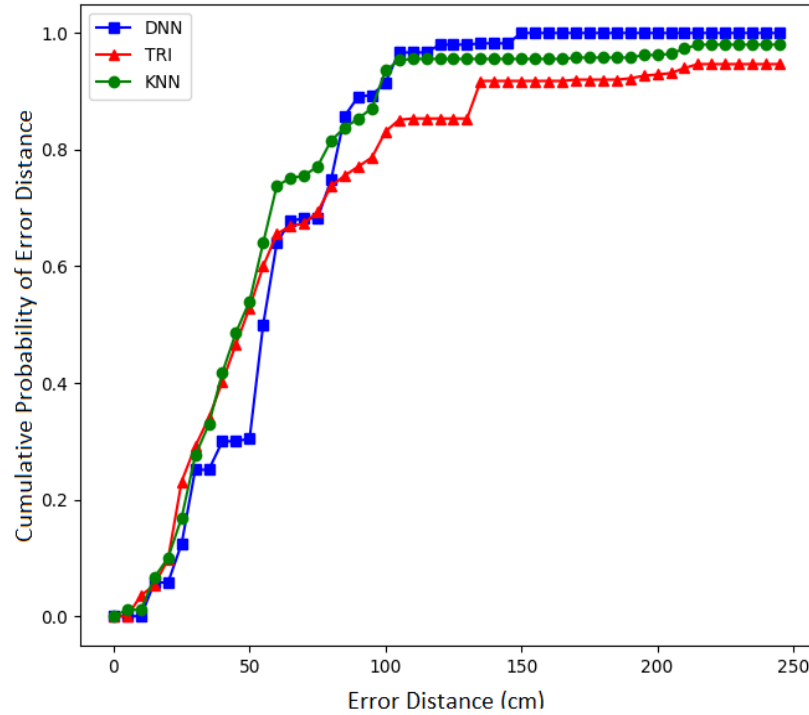


Figure 5.18: Cumulative Probabilities of Error Distances on all datasets combined of KNN only applied to all NNs' answers (Blue), only to all Trilateration's answers (Red), and all answers combined (Green)

As it was expected, using Dataset 1, the results in terms of cumulative probability of distance error show that, by using KNN outlier mitigation method, the lack of low distance error rate of the neural networks is compensated while maintaining the robustness on the higher end of distance errors.

Using all the Datasets combined, the results yielded may not be directly on par with the more sophisticated systems existing but maintains satisfying performances in most conditions. With NNs yielding average errors between 45 and 60 cm in its uncalibrated form, the proposed solution has much potential to improve when combined with further input refining algorithms and output enhancement methods such as a Kalman Filter.

Finally, once calibration was applied to the NNs, a significant boost in performances was observed in the system while applying KNN to sort through all the answers.

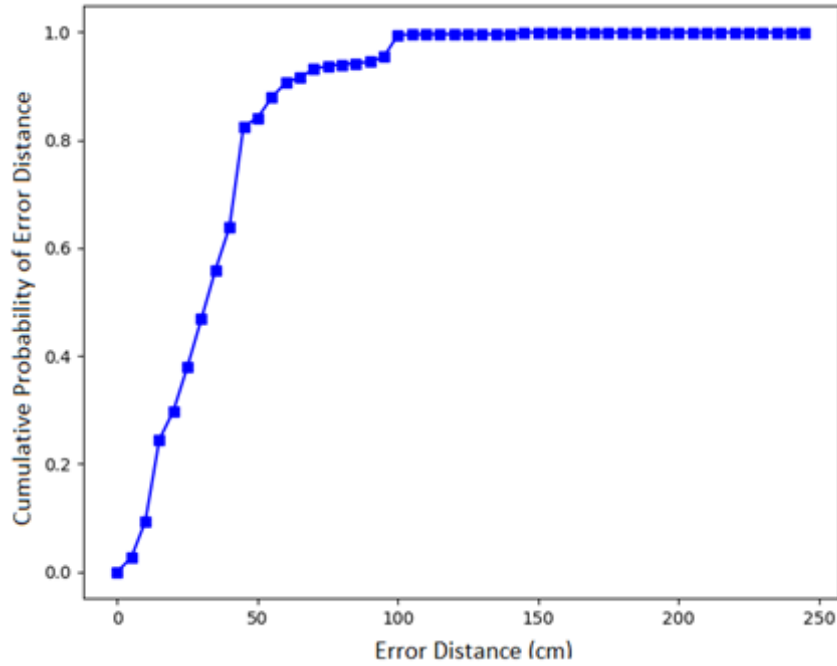


Figure 5. 19: Cumulative Probability of Error distance on 70% of all the datasets using KNN after calibration of the NNs using Fingerprint and Synthetic Data

NN Group	Mean Error (cm)	
	Dataset 1	Dataset 2+3
R1-R2-R3	40	51
R1-R2-R4	37	50
R1-R3-R4	41	48
R2-R3-R4	35	55
R1-R2-R3-R4	34	50
KNN	33	56

Table 5. 2: Average error of calibrated NNs

Figure 5.20 shows that calibration was successful, yielding noticeably better results with 80% of answers bellow 50 cm of error as opposed to 50% previously. The calibrated NNs were attained within 22 iterations of the calibration algorithm described in section 5.2, using batches of 1000 samples. The lower error of the NN using measurements from all 4 reference stations suggests that in cases with more reference stations available, the larger number of 4-Input NNs could provide a boost in performances.

5.2 Evaluation and Analysis

In the previous section it was shown that neural networks provide more consistent and accurate positioning than trilateration in poor measurement conditions even when trained only using synthetic data. The performances of the system were further improved after calibration, using only a set of 24 test points across the 13m x 9m lab in which the experiments were performed. They provided a better maximum error rate but were limited in terms of maximum accuracy. It remains however, that trilateration should be used if the presence of LOS and good measurement quality can be confirmed, but this remains a difficult/costly task as explained by the researchers at MIT from [15].

Additionally, KNN was introduced to sort through all the answers. This algorithm allows then to select only the best available cluster of answers to work with. In more sub-optimal signal conditions, it is expected that the neural networks answers will be more coherent and closer together than trilateration which will bias KNN's cluster selection. In the rarer cases of optimal conditions, trilateration answers can provide significantly more accurate results and will tend to form a better cluster even closer to the ground truth.

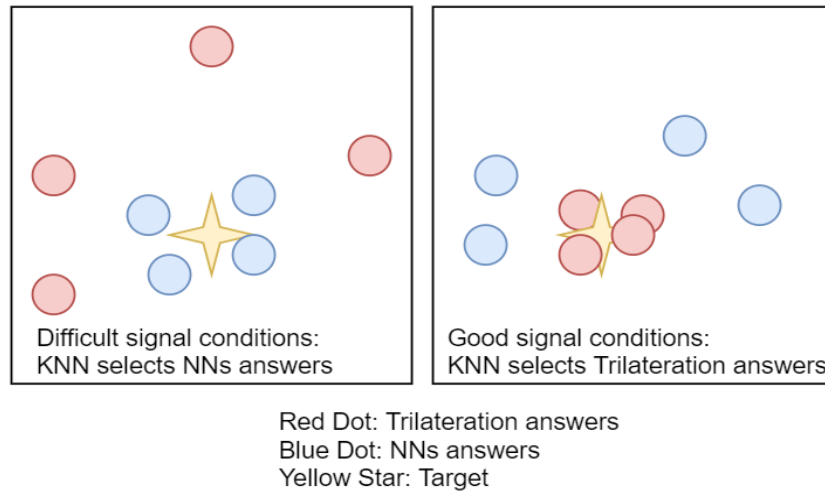


Figure 5. 20: KNN's cluster bias in Difficult vs Good measurement conditions (exaggerated example)

5.3 Discussion

Recent works such as in [35] can achieve accuracies of 10cm but require a minimum of 4 reference station's measurements and do not seem to use real data taken from a difficult environment; on the other hand, UWB-based IPS products on the market such as what *Infsoft GmbH* or Ubisense's *Dimension4*, claim to offer accuracies within 30cm on their website.

The system presented here demonstrated to be capable of accuracies of 35 cm in a simple environment conditions and 50 cm in a complex environment. While not on par with popular market solutions, the proposed solution did not cover the entirety of a IPS's process and was limited with rudimentary input correction where only absurd readings were discarded and the output was not enhanced, for example by using historical data such as the previous position, prediction algorithms or a Kalman filter. These weaknesses reveal the potential left unexplored of this thesis' work. Furthermore, it is important to realize that NNs require more powerful hardware to be run, this may limit the maximum number of tags that can be positioned within a given time frame. However, the cost of hardware accelerators for NNs such as GPUs has been significantly reduced in later years and APIs such as Keras already offer support functions to enable hardware acceleration. Additionally, in a larger coverage area, not all NNs would necessarily be run for every positioning request (not all readers may be able to make a measurement).

In conclusion, it can be said that KNN's effectiveness for indoor positioning was largely improved by using multiple NNs to support trilateration. This method achieved comparable performance to various works found in the IPS literature [31,32,33,36] but with overall less complexity for deployment and flexibility against system changes.

Chapter 6: Conclusion and Future Works

This thesis presented an indoor positioning system capable of accuracies around 35 cm and below 50 cm even in poor signal propagation conditions. It was shown that deep neural networks should be considered to replace or support to the strict mathematical formulations in positioning systems and that a swarm of less complex neural networks can rival more complex and data hungry ones. While trilateration allows for greater precision in optimal conditions, the distance error tends to spike rapidly as readings quality decreases. Neural networks offer greater flexibility when faced with incoherent data. The capability of making ranging measurements of UWB sensors allowed the use of synthetic data for training and stabilization of performance. The work in this thesis lead to an IPS capable of good accuracy when possible and robust positioning in less favorable conditions. Ultimately, the system is capable of accuracies rivaling systems found in the literature [31,32,33,36] while introducing more flexibility in deployment steps and operations. Unlike systems using a single neural network, adding a node does not require a full system retraining and the new network(s), depending on the environment, might begin useful operation based only on their synthetic data training as the results shown in the previous section show reliable results for uncalibrated NNs when using data collected in a good environment. In an actual integration, an important aspect to consider will be the selection algorithm that will manage which NN groups should be run based on the set of sensor data received. That is, NN groups should be run only if their inputs are valid and if only a subset of groups must be selected then an appropriate decision policy must be implemented. Validity of inputs can be determined by basic rules (zero, negatives and too large measurements can easily be discarded) and/or custom error mitigation algorithms that may weight quality of the measurements to select the best NN groups to run. A basic approach may involve prioritizing shorter measurements as they are less likely to have suffered from multipath and other interference sources.

Future works should aim at combining this positioning process with other existing techniques of input/output enhancement for localization systems such as Kalman filters and WCS. Also, more rigorous calibration of the hardware itself could have resulted in a boost in performance. The next phase of research should aim at exploring a similar concept using smaller cell sizes and a larger number of reference stations as well as using reference tags to collect real-time training data, rendering the system more adaptable to large changes in the environment. Additionally, a

hardware accelerated version of this project should also be developed to minimize training and execution times.

References:

- [1] L. Yu, M. Laaraiedh, S. Avrillon and B. Uguen, “Fingerprinting localization based on neural networks and ultra-wideband signals” IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Bilbao, pp. 184-189, 2011.
- [2] F. Serhan Daniş, A. T. Cemgil, “Model-Based Localization and Tracking Using Bluetooth Low-Energy Beacons” Sensors 17, 2017.
- [3] E. Mok, B. Cheung, “An Improved Neural Network Training Algorithm for Wi-Fi Fingerprinting Positioning.” ISPRS International Journal of Geo-Information. 2. 854-868, 2013.
- [4] J.-Y. Pyun, “Practical Fingerprinting Localization for Indoor Positioning System by Using Beacons.” Journal of Sensors, 1-16, 2017.
- [5] Cantón Paterna, V. Calveras Augé, A. Paradells Aspas, J. Pérez Bullones, “Bluetooth Low Energy Indoor Positioning System with Channel Diversity, Weighted Trilateration and Kalman Filtering.” Sensors, 2017
- [6] H. Mhaskar, T. Poggio, “Deep vs. shallow networks: An approximation theory perspective.” Analysis and Applications, 2016
- [7] P. Stephan, I. Heck, P. Kraus, G. Frey, “Evaluation of Indoor Positioning Technologies under industrial application conditions in the SmartFactoryKL based on EN ISO 9283.” IFAC Proceedings Volumes 13. 870-875, 2009.
- [8] W. Sakpere, M. Adeyeye Oshin, N. Boyfriend Wilton Mlitwa, “A State-of-the-Art Survey of Indoor Positioning and Navigation Systems and Technologies.” South African Computer Journal, 2017.
- [9] Y. Gu, A. Lo, and I. Niemegeers, “A Survey of Indoor Positioning Systems for Wireless Personal Networks,” IEEE Communications Surveys and Tutorials, vol. 11, no. 1, pp. 13–32, 2009.
- [10] J. Hightower and G. Borriello, “Location Systems for Ubiquitous Computing,” Computer, vol. 34, no. 8, pp. 57 – 66, 2001.
- [11] F. Seco, A. R. Jimenez, and F. Zampella, “Fine-Grained Acoustic Positioning with Compensation of CDMA Interference,” in IEEE Int. Conf. on Industrial Technology (ICIT), Seville, pp. 3418 – 3423, 2015.
- [12] F. Seco, J. C. Prieto, A. R. Jimenez, and J. Guevara, “Compensation of multiple access interference effects in CDMA-based acoustic positioning systems,” IEEE Transactions on Instrumentation and Measurement, vol. 63, no. 10, pp. 2368–2378, 2014.
- [13] J. C. Prieto, C. Croux, and A. R. Jimenez, “RoPEUS: A new robust algorithm for static positioning in ultrasonic systems,” Sensors, vol. 9, no. 6, pp. 4211–4229, 2009.

- [14] J. Tiemann, F. Schweikowski, and C. Wietfeld, "Design of an UWB Indoor-Positioning System for UAV Navigation in GNSS-Denied Environments," *International Conference on Indoor Positioning and Indoor Navigation*, pp. 1–7, 2015.
- [15] I. Guvenc, C. C. Chong, and F. Watanabe, "NLOS identification and mitigation for UWB localization systems," *IEEE Wireless Communications and Networking Conference, WCNC*, pp. 1573–1578, 2007.
- [16] L. Zwirello, T. Schipper, M. Jalilvand, and T. Zwick, "Realization Limits of Impulse-Based Localization System for Large-Scale Indoor Applications," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 1, pp. 39–51, 2015.
- [17] A. Cazzorla, G. D. Angelis, A. Moschitta, M. Dionigi, F. Alimenti, and P. Carbone, "A 5.6 GHz UWB Position Measurement System," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 3, pp. 675–683, 2013.
- [18] A. R. Jimenez and F. Seco, "Comparing Decawave and Bespoon 780 UWB location systems: indoor / outdoor performance analysis," *6th Int. Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 782 no. October, pp. 4–7, 2016.
- [19] Gogolak, Laslo et al. "Neural Network-based Indoor Localization in WSN Environments." *Acta Polytechnica Hungarica*, 10(6):221-235, 2013.
- [20] S. Kumar, S. Gil, D. Katabi, and D. Rus, "Accurate indoor localization with zero start-up cost," in *Proceedings of the 20th annual international conference on Mobile computing and networking*, pp. 483–494, ACM, 2014.
- [21] P. Davidson and R. Piche, "A Survey of Selected Indoor Positioning Methods for Smartphones" *IEEE Communications Surveys & Tutorials*, 2016
- [22]- S. Gezici, Z. Tian, G. B. Giannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu, "Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks," *IEEE signal processing magazine*, vol. 22, no. 4, pp. 70–84, 2005.
- [23] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single wifi access point," in *13th USENIX Symposium on Networked Systems Design and Implementation* 16, pp. 165–178, 2016.
- [24] 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.
- [25] B. Li, A. Dempster, C. Rizos, and J. Barnes, "Hybrid method for localization using WLAN," in *Proc. Spatial Sci. Conf.*, Melbourne, Australia, Sep. 2005, pp. 341– 350.
- [26] A. R. Jiménez Ruiz and F. Seco Granja, "Comparing Ubisense, BeSpoon, and DecaWave UWB Location Systems: Indoor Performance Analysis," in *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 8, pp. 2106-2117, Aug. 2017.
- [27] Bluetooth [Online]. Available: <http://www.bluetooth.com/>. [Accessed 13 August 2019].

- [28] Z. Liu, H. Darabi, J. Baneree, and P. Xu, "Wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Applications and Reviews)*, vol. 35, no. 6, pp. 1067–1080, 2009.
- [29] P. Cherntanomwong and D. J. Suroso, "Indoor localization system using wireless sensor networks for stationary and moving target," 2011 8th International Conference on Information, Communications & Signal Processing, pp. 1-5, 2011.
- [30] Y. Li et al, "Grid-based directed diffusion for wireless sensor networks", *Conference of Communications and Networking*, 2007.
- [31] L. Yu, M. Laaraiedh, S. Avrillon and B. Uguen, "Fingerprinting localization based on neural networks and ultra-wideband signals" 2011 IEEE International Symposium on Signal Processing and Information Technology, pp. 184-189, 2011.
- [32] W. Vinichayakul and S. Promwong, "Improvement of fingerprinting technique for UWB indoor localization" *The 4th Joint International Conference on Information and Communication Technology, Electronic and Electrical Engineering (JICTEE)*, Chiang Rai, pp. 1-5, 2014.
- [33] A. Taok, N. Kandil, S. Affes, "Neural Networks for Fingerprinting-Based Indoor Localization Using Ultra-Wideband." *Journal of Communications*, 2009.
- [34] C. Zhang, M. Kuhn, B. Merkl, M. Mahfouz and A. E. Fathy, "Development of an UWB Indoor 3D Positioning Radar with Millimeter Accuracy" *IEEE MTT-S International Microwave Symposium Digest*, pp. 106-109, 2006.
- [35] D. Ni, O. A. Postolache, C. Mi, M. Zhong and Y. Wang, "UWB Indoor Positioning Application Based on Kalman Filter and 3-D TOA Localization Algorithm" 11th International Symposium on Advanced Topics in Electrical Engineering (ATEE), Bucharest, Romania, pp. 1-6, 2019.
- [36] G. Caso, M. T. P. Le, L. De Nardis, M.-G. Di Benedetto, "Performance Comparison of Wi-Fi and UWB Fingerprinting Indoor Positioning Systems." *Technologies* 6. 14, 2018.