university of groningen

faculty of science and engineering

Non-linear Dimensionality Reduction for Indoor Localisation of Mobile Agents

Master's thesis

August 2019

Student: Sebastian Wehkamp

Primary supervisor: Dr. Kerstin Bunte

Secondary supervisor: Prof. Dr. Michael Biehl

ABSTRACT

In recent years an increasing amount of research has been done on the development of indoor positioning systems for mobile agents. Despite it being a challenging subject there is a large demand in services they can provide. An example of such a service, warehouse inventory management, is the service which inspired this thesis.

Outdoor positioning of mobile agents can be done using GPS which is not always available or precise enough for indoor positioning. Current research for indoor positioning systems focuses on approaches using cameras for visual navigation. This works with static indoor surroundings however the accuracy drops in dynamically changing environments, like warehouses. Additionally visual navigation methods are very sensitive to the lighting conditions which might not be optimal in all indoor environments causing the accuracy to drop drastically.

This thesis proposes several approaches which use dimensionality reduction to solve the problem. Wireless networks are present in most indoor environments so they are a cost effective approach. The high dimensional signal strengths are mapped to 2-dimensional grid positions. A new method for simulating wireless signal strengths using random paths is presented which resulted in a more realistic data set. Two novel extensions of the Self Organising Map algorithm are presented modifying the algorithm into either a supervised or a semi-supervised algorithm. Both of approaches were tested on the newly simulated data however their performance was unsatisfactory. A new technique was tested called non-linear t-SNE which provided very promising results. The ability to classify out-of-sample points quickly combined with the results of the dimensionality reduction resulted in one of the best performing algorithms so far. An extension of LVQ was tested and combined with the new method for data generation resulted in a very high accuracy for aisle classification.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

AoA Angle of Arrival

BLE Bluetooth Low Energy

dB Decibel

GMLVQ Generalised Matrix LVQ

GLVQ Generalized Learning Vector Quantization

GNSS Global Navigation Satellite System

GPS Global Positioning System

GTM Generative Topographic Map

IMU Inertial Measurement Unit

LGMLVQ Localised GMLVQ

LIDAR light detection and ranging

LVQ Learning Vector Quantisation

MCAR Missing Completely at Random

MIMO Multiple Input Multiple Output

PCA Principal Component Analysis

RSS Received Signal Strength

SLAM Simultaneous Localisation and Mapping

SNE Stochastic Neighbor Embedding

SOM Self Organising Map

TDoF Time Difference of Flight

ToF Time of Flight

t-SNE t-Distributed Stochastic Neighbour Embedding

1 INTRODUCTION

Warehousing in the 1990s differs completely from warehousing in the recent past. Back then warehouses were considered simple large stock-keeping entities and unavoidable cost centres. As a result of new business concepts and global competition this changed to the current situation where warehousing has become a critical activity in the supply chain in order to outperform competitors on customer service, lead-times, and costs [16]. Warehouses are now designed and automated for high productivity, high throughput rate, and low processing costs. This results in an increasing amount of processes being automated by mobile agents. In order for this to occur new methods and algorithms will have to be developed to localise these agents indoors. This is the premise of this thesis [21].

In order for a mobile agent to be able to work autonomously it should be able to reach the target location without human interference, in the literature, this is known as navigation. A key point for a mobile agent to navigate is to be able to localise itself. In outdoor environments the localisation can be done using a Global Navigation Satellite System (GNSS). An example of such a system would be the Global Positioning System (GPS). These systems rely on clear communication with satellites in order for an accurate signal to be received which is not always available indoors. Therefore this localisation method is often not available or precise enough to navigate complex indoor 3D environments. To localise the agent indoors other localisation methods are needed.

Current research on indoor positioning systems focuses on approaches using Simultaneous Localisation and Mapping (SLAM) in combination with cameras or more specialised hardware such as a light detection and ranging (LIDAR) sensor [2, 35]. The usage of sensors located on the agent itself for localisation can be classified as a relative positioning system. In contrast to these relative positioning systems there are absolute positioning systems which use external beacons for localisation. Algorithms have been and are currently being developed for both relative and absolute positioning systems. A brief discussion of these methods is listed in Chapter 2.

This thesis expands on absolute positioning techniques with a dimensionality reduction approach. Most indoor environments have beacons already present in the form of wireless networks which are used to assist the employees. This popularity of wireless communication equipment has greatly promoted indoor localisation services such as object tracking, and navigation. Popular methods for imple-

menting wireless network localisation is by using Time Difference of Flight (TDoF), Time of Flight (ToF), Angle of Arrival (AoA), or ultra-wideband networks. These implementations all require additional hardware and exact positioning of the wireless emitters which negates the advantage of being able to use the existing infrastructure. An economical alternative to these methods is a Received Signal Strength (RSS) based solution which have been studied in the past decades. These solutions do not require any additional hardware since the RSS value can be obtained with commodity hardware.

This thesis will focus on a solution which uses these RSS values and describe the localisation problem as a dimensionality reduction problem. The high dimensional input data consists of RSS values of all wireless emitters which are mapped to a low dimensional location in the warehouse. The goal is to have a mapping function which produces accurate outputs in near real-time. Near real-time computation is important because the mobile agent is moving while it performs localisation. If the localisation takes too long the mobile agent is already at another location and the calculated location is either obsolete or the mobile agent is required to move at a slower speed making the total solution less efficient.

Based on the described scenario and taking related approaches into consideration the research question this thesis aims to answer is:

Is it possible to determine the indoor location of mobile agents using RSS values of wireless networks with a dimensionality reduction approach?

The rest of this thesis will have the following structure; firstly related works are discussed in Chapter 2. The data is explained in Chapter 3, with an explanation of the methods and algorithms used in Chapter 4. The results are presented and discussed in Chapter 5. The thesis is concluded in Chapter 6 which also contains possible new research directions.

2

RELATED WORK

This section will cover the related research concerning indoor positioning systems. The indoor positioning research will be split into relative positioning systems and absolute positioning systems. Because the indoor localisation problem is described as a dimensionality reduction problem research covering dimensionality reduction will also be discussed.

2.1 RELATIVE POSITIONING SYSTEMS

Current research for indoor positioning systems focuses on using SLAM in combination with cameras or more specialised hardware like a LIDAR. This section will first discuss the Inertial Measurement Unit (IMU) in Section 2.1.1. The IMU is a sensor that measures forces acting on the agent and is used in conjunction with other sensors as it is not accurate enough on its own. The second sensor, which is discussed in Section 2.1.2, is the camera which can be placed in various configurations like a single camera, stereo camera, or an omni-directional configuration. The configuration used depends on the requirements. Section 2.1.3 discusses the third sensor, the laser sensor, which is more accurate than a camera and requires less computational power but does have a higher weight and power consumption. Lastly Section 2.1.4 discusses a way which allows an agent to localise itself using the earths magnetic field.

2.1.1 *Inertial measurement unit*

The IMU is a popular sensor for indoor localisation which measures forces acting on the mobile agent and can include accelerometers, gyroscopes, and magnetometers. They are popular because of their small size, low cost, and low power consumption. Most research [7, 38, 43] use an IMU as the basis to perform SLAM as these sensors do not have a high enough accuracy when used alone.

2.1.2 *Cameras*

In addition to an IMU a popular sensor is a camera which is used for computer-vision-based approaches. Cameras are popular because of their wide availability, light weight, low power consumption, and low price [54]. The resolution and framerate of the camera depends on the computational power available but most solutions have a resolution in

the range of 256px[38] to 1.6MP [7] and aim for 60 frames per second. The number of cameras differs based on the use-case. It is possible to perform SLAM using a monocular camera [20], a stereo camera [1], or in an omni-directional configuration [7]. Using a monocular camera for depth estimation requires a translational movement every time which is shown in Figure 2.1 [29]. This is not the case when using a stereo camera. If you allow for omnidirectional 4D movements (3D position and yaw) you will have to take obstacles in all directions into account which requires an omni-directional configuration. Increasing the number of cameras does increase the power consumption and the computational costs of processing the results, both of which might not be available at the agent, especially on flying agents. If you want to use the cameras for other purposes like dynamic object tracking or semantic labelling as well the computational costs increase even further. These visual navigation methods work well for relatively static indoor environments like office buildings [8, 13]. This is different from a warehouse where products, machinery, and people are constantly moving trough the environment. This could decrease the accuracy of these navigation methods severely. Another disadvantage is that cameras are very sensitive to illumination so they perform poorly in the dark. The warehouse in question is over 10 metres high with high shelves next to the aisles. This results in a large amount of low contrast areas closer to the bottom even when all of the lights are on. In addition to this the warehouse has an automated lighting system to reduce the power consumption but the result is that the lighting is not always on when the agent is in an aisle. The result is a severe drop in accuracy for visual navigation methods.

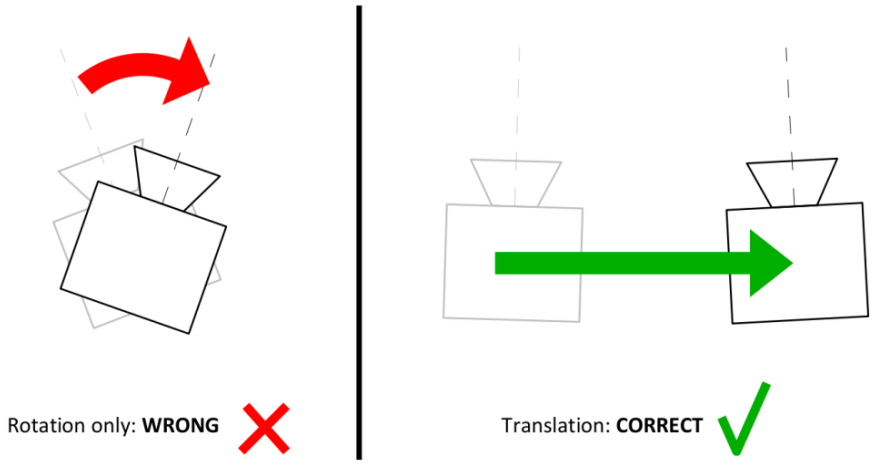


Figure 2.1: Translational movement which is required to initialise depth estimation using a monocular camera [29].

2.1.3 *Laser sensors*

The laser sensor has a number of advantages compared to computer-vision-based approaches. The main advantage being their increased accuracy at the cost of less computational resources. When combined with a mirror to deflect some of the beams downwards the laser sensor can also be used for accurate altitude estimation. The disadvantage of laser sensors is their increased weight and power consumption [43]. This can be seen when comparing for example an Hokuyo UST-20LX laser scanner and a Ximea mq013mg-e2 camera both used by [7]. The Hokuyo UST-20LX weights 130g and has a power consumption of 3.6W while the Ximea mq013mg-e2 weights 26 g and has a power consumption of 0.9W. The Hokuyo UST-20LX has a maximum detection distance of 60m and a scan speed of 25ms while having an accuracy of approximately 40mm. Mobile agents relying on just a laser scanner have the disadvantage that they cannot navigate straight corridor environments because of the lack of unique structural features and due to their maximum range. As an example, the previously mentioned Hokuyo UST-20LX has a maximum range of 60m. If an aisle in the warehouse is longer than 60m, which happens, it is hard to infer from the laser sensor information where inside the aisle the agent is located. To correct for this a monocular camera can be used to find unique features and to reliably perform loop closure [48].

2.1.4 *Magnetic field navigation*

Structures like pillars, metal constructions, and large objects refract earth's natural magnetic field. Magnetic field navigation is based on the idea to detect these irregularities using a magnetometer and use them as clues for finding the agent's location. The approaches require mapping a given indoor environment beforehand which means measuring the magnitude and direction of the magnetic field at each point. The agent can then find the most similar place in the magnetic field map to the one detected at a given point [23, 52].

A way to determine the location of the agent in the map is by estimating where the agent is most likely to be. This can be done by performing a maximum likelihood estimation which is a method that uses the given observations to estimate the parameters of the model, in this case the possible locations. The higher the value of a possible location, the more likely the agent is located at that location. The problem is that if many estimates have a high value the wrong location might be selected with a very large error. An alternative to maximum likelihood estimation is a particle filter which can select the optimal position even though many candidates have a high value. To update the weights of the particles three magnetic field maps are used, a horizontal intensity map, a vertical intensity map, and a direction

information map. The particles are propagated by using additional information like the current speed. In order to obtain this information the IMU can be used to know the robots forward velocity and angular velocity which allows to propagate the particles to the correct direction. However, the IMU has the problem of cumulative errors because every measurement depends on the previous one. Instead of using the IMU it is possible to use odometry. These velocity measurements do not depend on the previously measured velocity solving this problem but it does require a camera and more computational resources [27, 52].

Another approach to this problem presented by [23] is instead of determining the exact coordinates of the agent the goal is to identify the "room", or aisle in this use-case, in which the agent resides. Instead of creating a detailed magnetic map a signature for a given aisle is created. This signature is created by a random walk inside a given aisle. The frequency component of the magnetic signal is obtained using the Fourier transform of that signal. This method is independent of the exact path used when picking the magnetic signal making it robust [23, 52].

A disadvantage for both methods is that the map or signature has to be created beforehand. When large object move in the environment the magnetic map will change as well making the current map obsolete. This means that every time the environment changes the map or signature will have to be updated as well. This makes these methods more useful in static environments.

2.2 ABSOLUTE POSITIONING SYSTEMS

This section describes absolute positioning systems and covers three popular sensors used in such systems. Absolute positioning systems use external beacons to perform localisation of the mobile agent. An often seen use-case for absolute positioning systems is to compensate for the cumulative error relative positioning sensors suffer from. With relative positioning systems the current position estimate depends on the previous position estimate. This could result in an increasing error which is the cumulative error problem. This could be prevented if a relative positioning system is combined with an absolute positioning system improving the accuracy of the localisation. The main disadvantage of using an absolute positioning systems is that you have to place the localisation beacons throughout the environment which, depending on the size of the environment, can be a very cost and labour intensive job. First one of the older absolute positioning systems is discussed in Section 2.2.1, ultrasound. The equipment is relatively cheap but its very sensitive to noise making it unsuitable for some environments. Section 2.2.2 discusses Bluetooth localisation which became popular after the release of Bluetooth Low Energy (BLE) reducing the power consumption drastically. Lastly Section 2.2.3 discusses wireless

localisation which has the advantage that the wireless beacons, access points, are already present in most indoor environments.

2.2.1 *Ultrasound*

This type of absolute positioning system is popular because it is simple, inexpensive, and accurate. There are various methods explaining how such a positioning systems can be implemented but they all depend on the time of flight ToF. Examples of ultrasound positioning systems are the bat system [51], the cricket system [41], or the dolphin system [14]. The difference between these methods mainly being the distribution of the emitters and receivers, and the type of signal transmitted. In recent studies research has been done on using the time difference of flight TDoF instead of ToF. The advantage of using the TDoF is that it only measures the arrival times of signals thus it removes the need of synchronisation between the transmitters and receivers. A disadvantage of ultrasound positioning is the sensitivity to noise. In environments with a high amount of noise the accuracy of the positioning will drop or it might fail completely. Therefore using ultra sound positioning for the use-case at hand, a warehouse with heavy machinery, is not feasible [14].

2.2.2 *Bluetooth*

Since the release of BLE, an energy efficient version of Bluetooth, Bluetooth localisation turned out to be a very practical localisation method [50]. As an example indoor localisation tests were performed within the Computer Laboratory at the University of Cambridge where they concluded that even with sparse deployment of beacons there was an improvement in accuracy over localisation with wireless networks [22]. The recent release of Bluetooth 5.0 improves this localisation method even further since a maximum range of 400 meters can be obtained depending on the transmitting power and the environment. Because Bluetooth equipment is readily available on inexpensive commodity devices it is a very cost efficient localisation method. There are two ways to perform Bluetooth localisation, a low precision and a high precision version. The difference is the usage of the RSS indicator. In the low precision version the value of the RSS indicator is not used, the only concern is whether the receiver receives the signal or not. Because the maximum range of the transmitters is known a circle can be drawn around each transmitter. Once one of the signals is received that circle is highlighted. When more than one signal is received the overlapping region will be highlighted. In contrast to the low precision version, the high precision version does use the RSS value to scale the circles after which trilateration is used [50]. The RSS indicator value can be hard to obtain accurately because there are indications that wireless networks

cause errors in BLE RSS measurements. An incorrect RSS indicator can lead to wrong positioning [22].

2.2.3 *Wi-Fi*

Research on using wireless networks for indoor localisation dates back to 1998 when Microsoft presented Microsoft Research Radar [36]. At the time Microsoft used a combination of fingerprinting the RSS indicator values and dynamic changes like the recent movement history of the agent. Fingerprinting is still one of the state-of-the-art methods used for localisation and will be discussed below.

Several different techniques using wireless networks for localisation can be identified. The first technique is based on the RSS indicator which is measured for several different access points. This indicator is combined with a propagation model to determine the distance to the access point. Trilateration is then used to estimate the location of the device. This method is one of the easiest and cheapest to implement but it does not provide very good accuracy which is in the range of 2-4m. This is mainly due to the fact that in order to accurately perform trilateration exact knowledge of the beacon placement is required. Although the advantage of wireless localisation is that the beacons are already placed, the knowledge about the exact location of each beacon is often not which limits the accuracy. In addition to this objects where the wireless signal has to pass through cause the RSS indicator value to drop limiting the accuracy even further [55].

The second method is the method which Microsoft Research Radar proposed. It also uses the RSS indicator but combines it with a database of known coordinates and the respective RSS indicator values which is called a fingerprint. The database of fingerprints is created manually during an offline phase which can be very labour intensive depending on the size of the environment. In the online phase an unknown RSS indicator vector is matched against the database of fingerprints and the closest match is returned as the estimated user location. Such a system provides an accuracy in the range of 0.6-1.3m depending on the coverage of fingerprints. The main disadvantage is that you have to update the database of fingerprints when the environment changes [5, 56, 57].

The third wireless network localisation technique to be discussed is based on the ToF [34]. This approach uses timestamps provided by the wireless interfaces in order to calculate the ToF. The wireless signals propagation speed is close to the speed of light which remains constant in most propagation media. Therefore this method is not as susceptible to environment changes as the RSS indicator based techniques. The time stamps and the propagation speed are then used to estimate the distance to the wireless access points after which trilateration is used to estimate the position of the device. The result provides an accuracy of

2m. The difficulties for this approach are clock synchronisation, noise sampling artefacts, and multipath channel effects [34]. By using the two-way time transfer protocol the effect of the clock synchronisation error can be mitigated [28].

With the development of Multiple Input Multiple Output (MIMO) Wi-Fi devices, which use multiple antennas, it became possible to use the AoA of a signal received at the antenna array to apply triangulation and estimate the the location. For AoA of θ, the target's signal travels an additional distance of $d \cdot \sin(\theta)$ to the second antenna in the array compared to the first antenna. This results in an additional phase of $-2\pi \cdot d \cdot \sin(\theta) \cdot (f/c) \cdot (m-1)$ at the m^{th} antenna where f is the frequency and c is the speed of light. This situation is depicted in Figure 2.2 [32]. Though in principal the accuracy of this method is higher than the accuracy of the other methods it may require special hardware like an array of six to eight antennas [53] or rotating antennas [33]. The solution proposed by [32] called SpotFi proposes to use a super-resolution algorithm to accurately compute the AoA even when the access point only has three antennas. The result is a method with a median accuracy of 0.4m in a $160m^2$ area with five access points installed.

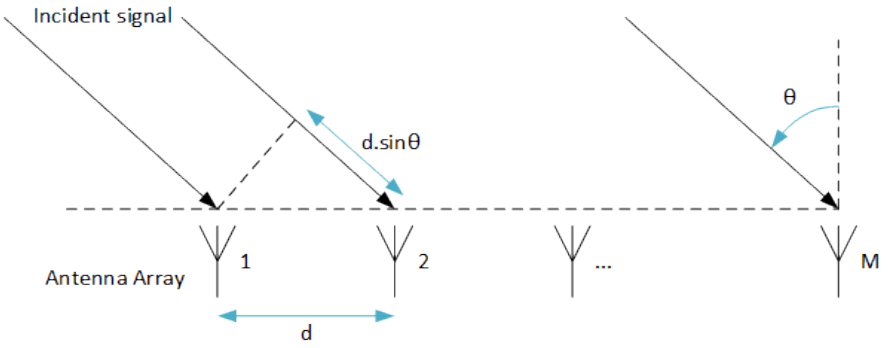


Figure 2.2: Uniform linear array available on MIMO wireless devices consisting of M antennas used for Angle of Arrival localisation [32].

2.3 SLAM

Most of the current research focuses on using a subset of the previously discussed sensors in combination with SLAM to perform the localisation and mapping. The simultaneous localisation and mapping problem describes a situation in which an autonomous agent starts in an unknown location in an unknown environment and then incrementally builds a map of this environment while simultaneously using this map to localise itself. Although this problem appears to be a chicken-and-egg problem a number of approaches have been proposed to address the problem [15, 18, 29].

SLAM has seen many developments over the past decades. These developments can be classified in two periods: a probabilistic period and an algorithmic analysis period. The probabilistic period is analysed in [6, 19]. This research covers the main probabilistic formulations for SLAM including the Extended Kalman Filters, Particle Filters, and maximum likelihood estimation which are all derivatives of the recursive Bayes rule. The main reason why probabilistic techniques are popular is the fact that the robot mapping is characterised by uncertainty and sensor noise. Probabilistic algorithms tackle these problems by modelling different sources of noise and their effect on the measurements [4]. The second period (2004-2015) focuses on the algorithmic analysis. Fundamental properties of SLAM are studied including observability, convergence, and consistency. The importance of sparsity for efficient SLAM solvers was understood and the main open-source SLAM libraries are developed [13].

2.4 DIMENSIONALITY REDUCTION

The relative positioning systems discussed in Section 2.1 are not suitable for the use-case at hand. Cameras suffer from bad lighting conditions in the warehouse while requiring a lot of computational resources, the performance of laser sensors drops drastically in straight corridor environments, and an IMU is not accurate enough to perform localisation on its own. Almost all of the absolute positioning systems discussed in Section 2.2 require exact knowledge of where the beacons are located, knowledge which is not always there. In larger environments it can be a large amount of work to collect this information. If dimensionality reduction in combination with wireless networks were to be used to solve the problem this exact knowledge is not required at all. The wireless access points are already available in most indoor environments and no extra knowledge is required. The idea is that the high dimensional RSS indicator values are mapped using some dimensionality reduction method to a 2-dimensional floor plan of the warehouse.

2.4.1 *Unsupervised methods*

Unsupervised methods do not have class information available to them and focus on distinguishing or grouping data into clusters based on their similarity to other members in the group. Due to the label information being unavailable to them they are cheaper in terms of data requirements compared to supervised methods. On the other hand, because of the absence of label information, they cannot self correct during their execution and have to rely on explicit or implicit assumptions.

Principal Component Analysis (PCA) is by far one of the most popular unsupervised algorithms for dimensionality reduction [40, 49]. The algorithm is based on finding orthogonal directions explaining as much variance of the data as possible. These directions are called the principal components. By using two principal components you would achieve the goal of dimensionality reduction. Although it is appealing because of its simplicity, this was already found in prior research [11] to be insufficient in the task of localising a mobile agent indoors.

A probabilistic approach commonly used for dimensionality reduction is the t-Distributed Stochastic Neighbour Embedding (t-SNE) algorithm developed by Maaten and Hinton which is the more popular variant of Stochastic Neighbor Embedding (SNE) [26]. t-SNE is a non-linear dimensionality reduction technique which aims to represent high dimensional data points as lower dimensional datapoints and position them such that similar points in the higher dimension are close in the lower dimension, while dissimilar points are positioned further away. A popular metric to measure (dis)similarity is Eculidean distance. Compared to SNE, t-SNE promises to be much easier to optimise and produce significantly better visualisations by reducing the tendency to crowd points together in the centre of the map [37]. The use of t-SNE for localisation was researched by [17] and they found that using t-SNE works well for some aspects of localisation however its reliance on Euclidean distance resulted in less than optimal results. [12] attempted to improve on this conclusion by using t-SNE in combination with the Mahalanobis distance and a newly developed relevance metric. The Mahalanobis distance is a distance measure that incorporates the feature-wise co-variance of the data. The newly developed relevance metric takes the changing importance of specific features towards the measurements within the dataset into account by including a relevance vector. The Mahalanobis distance performed better than the newly developed relevance metric and [12] concluded that the incorporation of the covariance of the data, suitably differentiates the data based on their positional information however it does not preserve the lane information very well. With both traditional techniques for dimensionality reduction providing insufficient results, different approaches must be considered.

Other unsupervised methods which were tested are k-medoids and the Generative Topographic Map (GTM) algorithm. K-medoids differs from the popular k-means algorithm [25] in that it does not update a mean prototype but takes the data point closest to the mean as the medoid. Another well researched unsupervised algorithm is the Self Organising Map (SOM) algorithm. SOM was introduced in [30] and allows for definition of a grid in a lower dimensional space with the approximate outline of the warehouse modelled in 2 dimensions. [11] found that SOM was limited by the winner-take-all approach in calculating similar points. Therefore a more probabilistic method was

deemed a better fit. [9] demonstrate an approach which overcomes these winner-take-all shortfalls by introducing such a probabilistic method called GTM. Similar to SOM it allows for definition of a grid in lower dimensional space however it allocates labels in a probabilistic manner instead of a deterministic manner. [12] concluded that GTM gave unsatisfactory results which is mostly due to the difficult initialisation of probabilistic methods. This is why a non-probabilistic method like SOM might still be better for this use-case.

2.4.2 *Supervised methods*

Supervised algorithms assign a position label to data based on the (dis)similarity of the trained prototypes. A popular classifier is the Learning Vector Quantisation (LVQ) algorithm [31]. Although the LVQ algorithm originally was not designed to be used for dimensionality reduction but as a classifier it can be used for dimensionality reduction. Its extensions allow for interpretable dimensionality reduction and visualisation which we will focus on in this thesis. The algorithm has been extended several times over the recent years where the most important improvement is the development of Generalized Learning Vector Quantization (GLVQ). GLVQ introduced a cost function which turned the problem into an optimisation problem. With the introduction of a cost function the problem could now be optimised using stochastic gradient descent [42]. The perfromance of GLVQ showed improvements compared to LVQ however it still relied on Euclidean distance which has been proven to be unsuitable for the task [12]. Due to the complexity of the data the use of an appropriate distance measure is important to get an adequate representation of the data. Relevance learning techniques extend e.g. the Euclidean distance with weight factors for the different dimensions. These weight vectors are optimised together with the prototypes during the training phase. This approach however ignores correlative effects between features. The Generalised Matrix LVQ (GMLVQ) [46] introduces a full matrix of relevance factors in the distance measure. This was further extended to not only account for global relevance, but also local relevances based on each prototype or class which is called Localised GMLVQ (LGMLVQ) [44]. [12] discusses both GMLVQ and LGMLVQ in their research. They concluded that both GMLVQ and LGMLVQ outperform the unsupervised alternatives because of their self-correcting nature and the best performing combination was using LGMLVQ with random initialisation.

3 DATA AND PREPROCESSING

This chapter describes the data which was collected and simulated for this research. In order to obtain the data in a short amount of time a simulation similar to [12] was used. To remain suitable for the research it was made sure that the simulated data was similar to the real world data.

3.1 DATA COLLECTION AND SIMULATION

The data used in this thesis consists of RSS indicator measurements at various locations in the warehouse. These measurements are the higher dimensional data which should be mapped to a 2-dimensional floor-plan of the warehouse. To make the generated data as similar as possible to the real world data, features of the real world had to be simulated. These features are based on where the real world data was gathered in [11, 17]. The real world data was gathered in four aisles of the warehouse with a total width of 12m, a maximum height of 5m, an aisle length of 60m, and an extra 2m connecting the aisles. An image of the floor plan can be found in Figure 3.1. A visual depiction of the simulated warehouse can be found in Figure 3.2a.

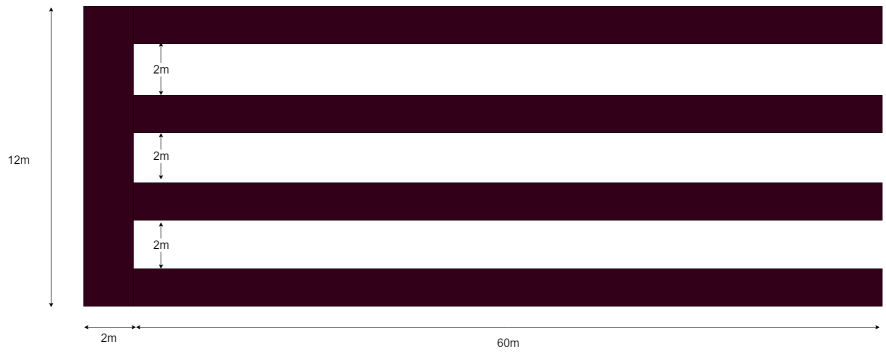
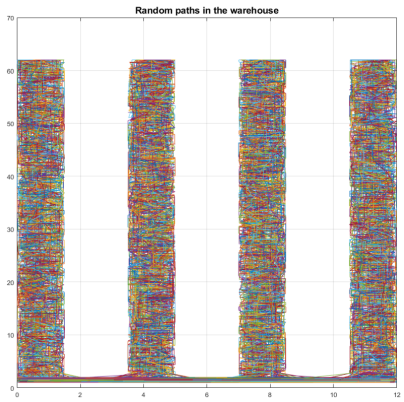


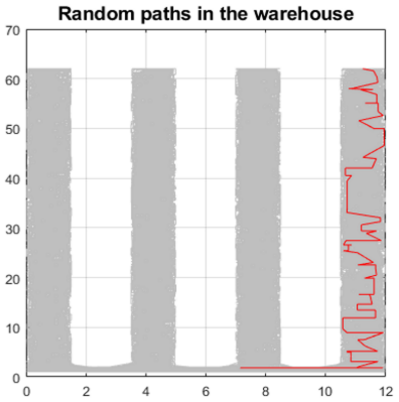
Figure 3.1: Figure depicting the dimensions of the investigated part of the warehouse. The warehouse consisted of four equally separated aisles and a pahthway aisle.

The locations from which to take the simulated measurements are generated using random paths in order to make the scenario more realistic and as it allows for the inclusion of time-series information in the future. At the end of each aisles is the starting point for the random paths and the end goal is in the bottom left. In every aisle 100 paths start and they end either by reaching the end goal, or by reaching their max length of 200. The paths can go either left, right, or

down. The step size of the random paths is randomly generated per step. The result is a data set of random locations inside the aisles and the connecting aisle. The random paths are shown in Figure 3.2a and a single path is highlighted in Figure 3.2b. The positions of the data points together with the label information is shown in Figure 3.3.



(a) Figure depicting the simulated dimensions of the warehouse. The lines are the random paths along which the measurements were taken for the simulated data.



(b) A single path is highlighted in red to show how it randomly moves through the aisle.

Figure 3.2: The figure depicts how the random path generation algorithm works.

The dataset of positions is then used to calculate the signal strengths. This is done in the same way as [12], using an industrial standard defined in [47], with the function:

$$L_{total} = L(d_0) + N \log_{10} \frac{d}{d_0} + L_f(n)$$

in which N is the distance power loss coefficient, f the frequency (MHz), d the separation distance (m) between the base station and portable terminal (where $d \geq 1$ m), d_0 the reference distance (m), $L(d_0)$ the path loss at d_0 (Decibel (dB)). For a reference distance d_0 of 1 m and assuming free-space propagation this can be calculated with $L(d_0) = 20log_{10}f - 28$ where f is in MHz. L_f represents the floor penetration loss factor (dB) and n the number of floor between the base station and portable terminal [12, 47]. Since L_f represents the loss factor of multiple floors and the warehouse is a single floor environment we can assume n and thus L_f to be 0 and leave it out of the equation. The frequency of the wireless network in the warehouse is 2.4GHz (2400MHz) and we adjust the value of N as documented in [47]. This was done by [12] until the simulated L_{total} values resembled the real world data. The final value $N = 20$ was chosen and the generated dataset was compared to the real world data in order to determine if the simulated data can be used to test the various methods.

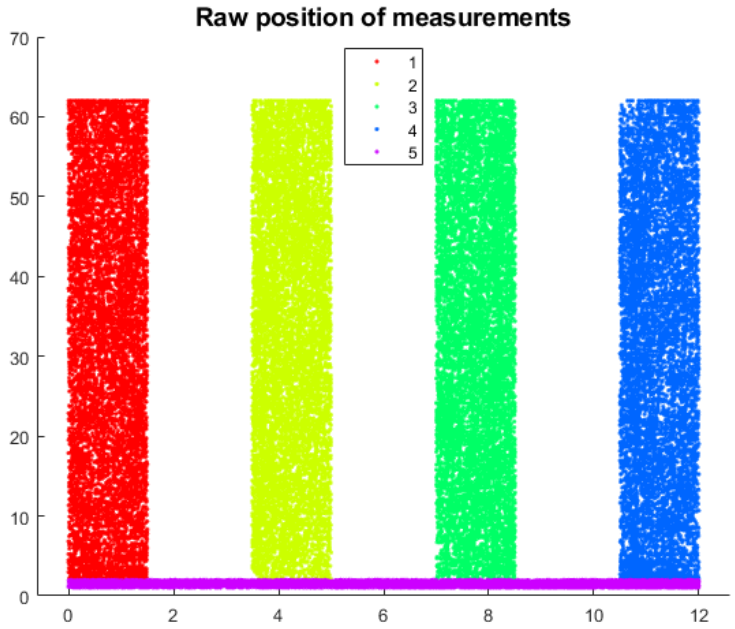


Figure 3.3: Figure depicting the simulated data positions together with label information.

Compared to the real world data a notable difference is the absence of missing data in the generated dataset. This removes the need for imputation methods. Although it may have been possible to add missingness to the data, it was seen as an additional complexity which is out of the scope of this project. In addition to this, according to [11, 12, 17] the missing values in the real world data were not fully Missing Completely at Random (MCAR) which is a requirement for many imputation methods.

3.2 DATA DESCRIPTION

The raw data is a combination RSS indicator values and the 3-dimensional location in the warehouse. There are 39 wireless access points in range which are located across the environment. The location will act as the class label for validation purposes.

The data set contains 80400 records, the statistical analysis of which can be found in Table 3.1. Figure 3.4 shows the box plots of the individual features. Although a slightly different data generation method has been used compared to [12] from both Table 3.1 and Figure 3.4 we can conclude that the data is very similar and so the test results are comparable. The statistical information of all of the individual features can be found in Table A.1 in Appendix A.

	Mean	Median	Std. deviation
1	0.5224	0.4753	0.0229

Table 3.1: The mean, median, and standard deviation values for the simulated data set.

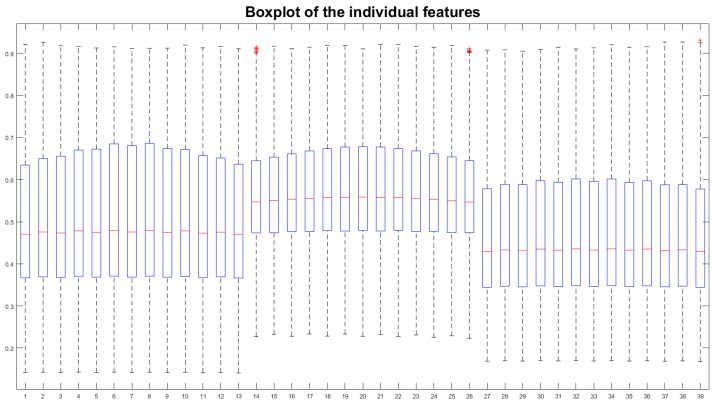


Figure 3.4: Boxplots of the 39 individual features containing the median and the range for the simulated data as well as the outliers.

3.3 PREPROCESSING

As stated previously, we do not have to repair the data since missingness was not simulated. Therefore there is no need for imputation methods.

The most important preprocessing step performed is identical to [12]. We scale the signal strengths x from dB values ranging from -90 to 0 to a new scale. In the old scale 0 represents the strongest possible signal strength and -90 the lowest possible. This is transformed to a scale between 0 and 1 in which 0 is the lowest possible signal strength and 1 the highest. This can be done using either a linear or a non-linear mapping. [12] tested both methods and concluded that the non-linear transformation provided the best results so we will be using that throughout this thesis. The formula for the transformation is:

$$y = \frac{L}{1 + e^{-k(x-x_0)}}$$

which can be seen in Figure 3.5. x is the input signal strength, x_0 is the midpoint which in this case is set to 45, and $k = 0.1$ as recommended by [12].

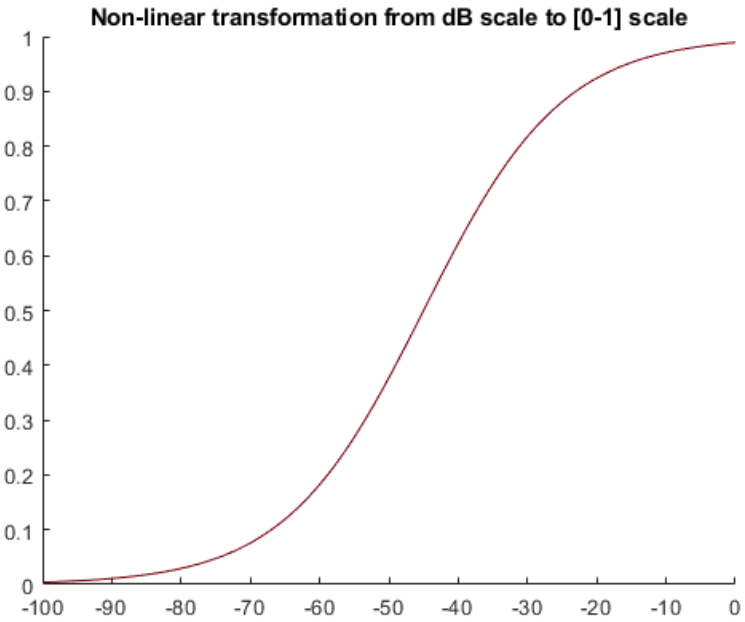


Figure 3.5: Figure depicting the non-linear transformation used to scale the RSS indicator values from ranging between -90 and 0 to a range between 0 and 1. The value of $k = 0.1$ is used as suggested in [12].

After the scaling is done a z-score transformation is applied in order to compare the measured values with respect to their relative position in the distribution. This is done using the equation:

$$z_{if}^t = \frac{y_{if}^t - \mu_f^t}{\sigma_f^t} i = 1...n, f = 1...F$$

in which y_{if}^t is the value of the scaled feature in record i of the dataset t and μ_f^t and σ_f^t are the corresponding mean and standard deviation of the data t. This transform is applied to the full data set after which the algorithms discussed in Chapter 4 are applied. All of the preprocessing steps are similar to [12] in order to be able to compare results.

3.4 MAGNETIC FIELD INFORMATION

To improve upon the result of [12] we aimed to incorporate one of their future work suggestions, incorporate magnetic field information. As stated in Section 2.1.4, the idea is that large metal structures interfere with the earths geo-magnetic field. These interferences can be detected and this could introduce another dimension in the data with a new and different characteristic compared to the wireless network information. Especially in a warehouse with a lot of repeating metal structures these interferences should be present and could be used to identify boundary points between two positions.

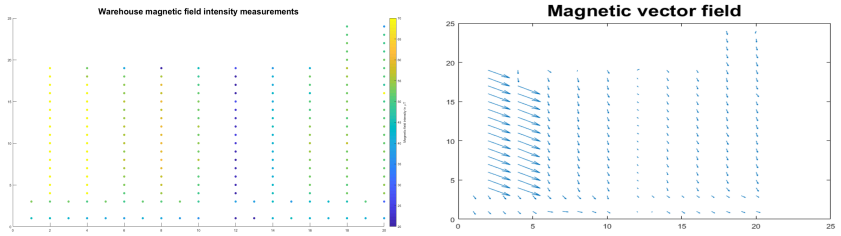
The plan was to gather the information by using a custom made application for an android device. This application then measures the direction and the strength of the magnetic field and transfers it to a SFTP server. The tablet with the application would be mounted on a forklift and run the application for a period of two weeks. In order to match the magnetic field information to a location the log data from the warehouse inventory system will be used. The warehouse inventory systems knows when a product is collected, the timestamp, and the location of the product. This information can be merged with the data from the Android application in order to obtain a magnetic field map. Problems arose when installing the application on the tablet (Samsung Galaxy Tab A 10.5 WiFi 2018). The first problem was that in order to specify the SFTP server, the server which receives the data, the application has to be recompiled. This had to be done by using an older version of Android Studio, on an older Ubuntu version, with older plugins. Some modifications were still necessary to the source code in order to be able to compile it. When this succeeded the application was tested on a smartphone (Samsung Galaxy S7, Android 8) and it was running successfully. When the application was installed on the tablet which was to be mounted on the forklift, the application ceased to run. It only transferred the data on application start up and then stopped working. This is probably due to the tablet running the latest Android, Android 9. In the newer versions of Android, starting from Android 9, Google severely limits the number of Wi-Fi scans to four scans every two minutes for foreground apps and one scan every 30 minutes for background applications [3].

	Mean	Median	Standard deviation
Magnetic field data	52.0309	77.1780	73.8519

Table 3.2: The mean, median, and standard deviation values for the magnetic field intensity (μT).

To get an idea of what the magnetic field information would look like we collected it manually using a different application on the tablet. Manual measurements were taken in five aisles on the left of the warehouse, three aisles in the centre, and two longer aisles in the centre inside a concrete bunker. The intensities at the locations are shown in Figure 3.6a and the statistical analysis can be found in Table 3.2. From Figure 3.6a you can conclude that the magnetic field caries a lot of information separating the aisles. The left two aisles have a higher magnetic field intensity overall than the others while the most centrally located aisle has the lowest magnetic field intensity. Even within the aisles there is some variance in intensity which might be useful for localisation. The direction of the vector field is consistent as can be seen in Figure 3.6b with only some slight

differences between the aisles. Within the aisles the direction is mostly consistent as well with the exception of the aisle end. This makes sense since the aisle end is located against a warehouse wall. We also measured the magnetic field along the width of an aisle, the intensities can be seen in Figure 3.7a and Figure 3.7b. The direction is consistent over the width of the aisle while the intensity differs slightly with a higher intensity on the left of the aisle.



(a) Figure depicting the magnetic field intensity (μT). The five aisles on the left are located on the left inside the warehouse. The aisles after the gap located at 11 are located in the centre of the warehouse. The two longer aisles are aisles inside a concrete bunker in the warehouse.

(b) The figure shows the direction of the magnetic field.

Figure 3.6: Magnetic field measurements taken throughout the warehouse at fixed locations.

These figures are based on relatively small set of measurements in order to get an idea of what the magnetic field data looks like however the dataset size is not large enough in order to be used in the algorithms. Since fixing the Android application is considered out of the scope of this project this was not attempted. There are some suggestions for improvements of the application besides having a working application on Android 9 which will be listed here.

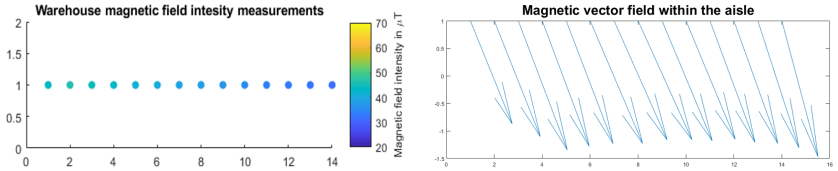
1. Find a solution for the afore mentioned limitation imposed by Android 9 and newer.
2. Specifying the SFTP server in the application settings instead of hardcoding the server location.
3. Allow for local storage in case the connection to the server is unavailable.
4. Show the individual vector components of the magnetic field instead of just the intensity and transfer this data as well.
5. Graph the data live inside the app. This would be interesting for manual testing purposes.
6. Specify the measurement timing interval. This is related to the next point.

7. Improved battery saving. At this moment the application drains the battery but it should preferably be able to run an entire work day (8 hours) while running the application in the background.

8. Show the orientation of the tablet in order to allow for consistent measurements.

9. A manual measurement mode for testing purposes.

This should allow to gather the magnetic field data in a consistent matter.



(a) The figure shows the 2-dimensional direction of the magnetic field along the width of the aisle.

(b) The figure depicts the 2-dimensional direction of the magnetic field at various measurement location in the warehouse.

Figure 3.7: Both figures depict magnetic field information measured along the width of an aisle.

4 METHODOLOGY

Algorithms which use Euclidean distance as a metric are insufficient for the task at hand as found by [11, 12, 17]. Therefore other methods and different metrics have to be considered. [12] found that using the Mahalanobis distance resulted in better results compared to Euclidean distance. This chapter discusses the various supervised, unsupervised, and semi-supervised methods and algorithms chosen to perform localisation inside the warehouse.

4.1 UNSUPERVISED METHODS

Unsupervised methods are those which focus on separating the data based on the similarity to other members of the group. This is done without the label information being available to the algorithms during execution. The result is that these algorithms are cheaper in terms of data requirements compared to supervised methods however they cannot self-correct during the algorithm execution.

4.1.1 *SOM*

A SOM consists of neurons r organised on a regular low-dimension grid. Each neuron has a d-dimensional weight vector w attached (prototype vector). d is equal to the dimension of the input vector. The neurons on the regular grid are attached to the adjacent neurons r' by a neighbourhood relation $h_\sigma(r, r')$. Example structures are a hexagonal or a rectangular relationship. During the training of SOM the weight vectors are moved such that they span across the input vectors v until the map is organised. Neighbouring neurons on the grid have similar prototypes [30].

The training process follows a stochastic gradient descent using the cost function [10]:

$$E_{SOM} = \sum_r \delta_{r,s(v)} \sum_{r'} h_\sigma(r, r') \cdot d(v, w_{r'})$$

where δ is the Kronecker delta symbol, d a distance metric, and $s(v)$ is:

$$s(v) = \arg\min_{r \in A} \left(\sum_{r' \in A} h_\sigma(r, r') \cdot d(v, w_{r'}) \right)$$

and the neighbourhood relationship is defined as:

$$h_\sigma(r, r') = \exp\left(\frac{-\mathcal{N}(r, r')}{\sigma} \right)$$

After derivation the update rule is defined as:

$$\Delta w_r = -\xi h_\sigma(r, s(v)) \frac{\partial d(v, w_r)}{\partial w_r}$$

In which ξ is the learning rate for stochastic gradient descent.

4.2 SUPERVISED METHODS

Supervised methods are methods in which the algorithm has access to the label information during execution of the algorithm. These algorithms focus on separating the classes using this information. Since they have access to the additional information they are more costly with respect to the data requirements as the labels have to be obtained.

4.2.1 *LGMLVQ*

As mentioned in Chapter 2 the GMLVQ improves on the GLVQ algorithm. This is done by replacing the traditional Euclidean distance metric with adaptive similarities. The cost function of GMLVQ can be represented as [45]:

$$\mathrm{E}_{\mathrm{GMLVQ}} = \sum_i \Phi(\mu_i) \text{ where } \mu_i = \frac{d_J^\lambda(\xi_i) - d_K^\lambda(\xi_i)}{d_J^\lambda(\xi_i) + d_K^\lambda(\xi_i)}$$

based on the steepest descent method. Φ is a monotonic function like the identity function $\Phi(x) = x$, $d_J^\lambda(\xi_i) = d^\lambda(w_J, \xi_i)$ which is the distance of data point ξ_i to the closest prototype w_J with the same class label y_i, and $d_K^\lambda(\xi_i) = d^\lambda(w_K, \xi_i)$ is the distance from the closest prototype w_K with a different class label than y_i.

The LGMLVQ algorithm is an extension on the GMLVQ algorithm. LGMLVQ uses an extended model which uses local matrices Λ attached to each prototype. The result is a model which allows for non-linear decision boundaries instead of piecewise linear boundaries as in GMLVQ [45]. The cost function of LGMLVQ is very similar to the cost function of GMLVQ as can be seen below.

$$\mathrm{E}_{\mathrm{LGMLVQ}} = \sum_i \Phi(\Psi^i_{\mathrm{Local}}) \text{ where } \Psi^i_{\mathrm{local}} = \frac{d_{\omega_J}^{\Lambda^{\omega_J}} - d_{\omega_K}^{\Lambda^{\omega_K}}}{d_{\omega_J}^{\Lambda^{\omega_J}} + d_{\omega_K}^{\Lambda^{\omega_K}}}$$

Similar to the GMLVQ cost function $d_{\omega_J}^{\Lambda^{\omega_J}}$ and $d_{\omega_K}^{\Lambda^{\omega_K}}$ are the closest prototypes w with the same and a different class label y_i respectively. $d^{\Lambda^j}(x^i, w^j)$ is calculated using the local matrices as:

$$d^{\Lambda^j}(x^i, w^j) = (x - w^j)^\intercal \Lambda^j (x - w^j) \text{ with } \Lambda^j = \Omega^{j\intercal} \Omega^j$$

This means that LGMLVQ not only learns the prototypes w_j but also the corresponding matrices Ω_j. The relevance matrix Λ can be calculated as:

$$\Lambda^j = \Omega^{j^T} \Omega^j$$

The final algorithm is shown in algorithm 1.

Algorithm 1: LGMLVQ algorithm

Initialise prototypes w^j;
Initialise matrices Ω^j and normalise such that $\sum_{ik}(\Omega_{ki})^2 = 1$;
while *stopping criteria not reached* **do**
  randomly select training sample x^i;
  compute the distances $d^{\Lambda^j}(x^i, w^j)$ to the prototypes w^j;
  determine closest correct $w^j = \arg\min_j d^{\Lambda^j}(x^i, w^j)$ with $y^i = c(w^j)$;
  determine closest incorrect $w^k = \arg\min_j d^{\Lambda^j}(x^i, w^j)$ with $y^i \neq c(w^j)$;
  update the prototypes according to $w^L \leftarrow w^L - \tau_1 \cdot \frac{\partial E_{LGMLVQ}}{\partial w^L}, L \in J, K$;
  update the matrices according to $\Omega^L \leftarrow \Omega^L - \tau_2 \cdot \frac{\partial E_{LGMLVQ}}{\partial \Omega^L}$;
  normalise the matrices such that $\sum_{ik}(\Omega_{ki})^2 = 1$;
end

[12] concluded that both GMLVQ and LGMLVQ perform best when randomly initialised instead of initialising with the covariance of the data. They also concluded that LGMLVQ performs better than GMLVQ and therefore the best approach is to use LGMLVQ with random initialisation.

4.2.2 *Local linear t-SNE Mappings*

Most traditional dimensionality reduction techniques provide a mapping of the given input data points only, instead of an explicit mapping function. The result is that additional effort has to be done to consider out-of-sample data points. [10] proposes a general principle to obtain such a mapping function. The principle is based on defining an explicit mapping function:

$$f_w : \mathbb{R}^D \to \mathbb{R}^d, x \to \hat{y} = f_w(x)$$

[10] proposes to fix a parameterised form of f prior to training. The parameters are then optimised according to the objective of the respective dimension reduction method, which in this case is parameterised t-SNE. Parameterised t-SNE extends normal t-SNE by using a multilayer

neural network. The parameters of the network are then optimised using the cost function:

$$E_{t-SNE} = \sum_i \sum_j p_{ij} \log \left(\frac{p_{ij}}{q_{ij}} \right)$$

in which p_{ij} is the conditional probability that the high dimensional point x_i would pick x_j as its neighbour. Similarly q_{ij} is the conditional probability that the low-dimensional counterparts y_i and y_j would be neighbours. This means:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n} \text{ and } q_{ij} = \frac{(1 + \frac{d_\varepsilon(\varepsilon^i, \varepsilon^j)}{\varsigma})^{-\frac{\varsigma+1}{2}}}{\sum_{k \neq l} (1 + \frac{d_\varepsilon(\varepsilon^k, \varepsilon^l)}{\varsigma})^{-\frac{\varsigma+1}{2}}}$$

It is assumed that the local linear projections have the form:

$$x^l \to p_k(x^l) = \Omega_k x^l - w^k$$

where Ω_k are the local matrices and w^k are the prototypes. These local pieces can be combined with:

$$f_w : x \to \hat{y} = \sum_k r_{lk} (L_k \cdot p_k(x^l) + l^k)$$

In which r_{lk} are the responsibilities of mapping p_k to x^l. This could for example be Gaussians centred around the points. It is assumed that $\sum_k r_{lk} = 1$. l_k are the local offsets used to align the local projections L_k. The number of parameters W depends on the number of local projections k and their dimensions m. Generally the number of parameters is smaller compared to projecting all points y^i directly. Therefore it is sufficient to consider only a small part of the training data to obtain a valid dimension reduction. The parameters are optimised using stochastic gradient descent based on the derivative of the t-SNE cost function [10].

4.2.3 *Supervised SOM*

In its original form, as described in Section 4.1.1, SOM is an unsupervised algorithm. This section presents a way of using supervised information obtained using LGMLVQ in SOM. Instead of using standard dissimilarity metrics like Euclidean distance or Mahalanobis distance the dissimilarity metric of GMLVQ and LGMLVQ is used. The metric is as stated earlier:

$$d^\Lambda(x, w) = \sum_{ijk} (x_i - w_i) \Omega_{ki} \Omega_{kj} (x_j - w_j)$$

The derivative of which yields:

$$\nabla_w d^\Lambda(x, w) = -2\Omega^\intercal \Omega (x - w)$$

The update rule is then defined as:

$$\Delta w_r = -\xi h_\sigma(r, s(v)) \cdot -2\Omega_r^\intercal \Omega_r (v - w_r)$$

In which Ω_r is the local matrix belonging to the LGMLVQ prototype most similar to the data point.

4.3 SEMI-SUPERVISED METHODS

Semi-supervised methods are methods which are in between supervised and unsupervised learning. In addition to using unlabelled data, the algorithm is provided some supervised information for a small amount of examples. In this case a data set $X = (x_i)_{i \in |n|}$ can be divided into two parts: $X_l := (x_1, ..., x_l)$ for which labels $Y_l := (y_1, ..., y_l)$ are provided, and $X_u := (x_{l+1}, ..., x_{l+u})$ for which no labels are provided. There are other forms of semi-supervised learning possible like a constraint such as "these points do not have the same target", this however will not be used in this section. In comparison to a supervised algorithm which uses only labelled data a semi-supervised algorithm which takes also the unlabelled data into account can have a more accurate prediction. However for this to occur [39] states that there is an important prerequisite. The distribution of examples, which the unlabelled data will help elucidate, must be relevant for the classification problem. In mathematical terms, the knowledge $p(x)$ which one gains through unlabelled data must be useful in the inference of $p(y|x)$. If this is not the case it might even happen that the performance degrades when using unlabelled data since it is misguiding the inference [39, 58].

4.3.1 *Semi-supervised SOM*

This section presents an extension of SOM which allows for semi-supervision. We modify the SOM cost function by adding a supervised term to that cost function. This term represents a sum over samples of which we know which part of the SOM map should represent them. The modified cost function is:

$$E_{SOM} = \sum_r \delta_{r,s(v)} \sum_{r'} h_\sigma(r, r') \cdot d(v, w_{r'}) + \sum_{i=l+1}^{n} (w_i - p_i)^2$$

In which p represents the true corresponding signal strength for that position. So the larger the distance of w_i to the known correct location, the larger the contribution of that term will be to the final cost. The second term will be 0 if no true signal strength p is known. The derivative of the new cost function yields the following update rule:

$$\Delta w_r = -\xi \cdot (\alpha(h_\sigma(r, s(v)) \cdot (v - w_r)) + \beta(w_r - p_r))$$

Where α and β are the the contributions of the unsupervised and supervised term respectively. The supervised term is only > 0 for the closest weight if p is known. For this reason it is also possible to use the neighbourhood influence for the supervised term as well. The update rule for this situation is:

$$\Delta w_r = -\xi \cdot (\alpha(h_\sigma(r, s(v)) \cdot (v - w_r)) + \beta(h_\sigma(r, s(v)) \cdot (w_r - p_r))$$

Which causes more prototypes to move when a true signal strength is known.

5 RESULTS

This chapter will present,describe, and discuss the results obtained using the methods described in Chapter 4. First the results the unsupervised methods are discussed in Section 5.1 followed by the results of the supervised algorithms in Section 5.2. The chapter is concluded with a discussion of the semi-supervised results in Section 5.3.

5.1 UNSUPERVISED METHODS

There are traditional dimensionality reduction methods available some of which were tested by [12]. For comparison reasons and to see whether these methods would suffice it is attempted to reproduce the results of [12].

5.1.1 *SNE*

The first unsupervised algorithm tested was SNE. Figure 5.1 shows the converged part of the data mapped using SNE combined with the mahalanobis distance and a perplexity of 50. There are four outliers scattered on the outside (not shown in Figure 5.1) which indicate that the algorithm has not converged completely. Different perplexities were tried but none resulted in convergence. In the converged part some positional information remains. The relative location of the pathway aisle is correct with respect to the other four aisles. Furthermore the data does have a shape somewhat similar to an aisle. The order of the aisles is going from the inside to the outside with the first aisle being in the centre and the last aisle being on the outside. The final lower dimensional mapping using SNE does not appear to be suitable for the use-case at hand.

5.1.2 *t-SNE*

Since the results of SNE were unsatisfactory t-SNE was tried in combination with various distance measures which is discussed in this section. [12] found that using t-SNE with Euclidean distance did not work as well and using the mahalanobis distance provided much better results. The result of t-SNE with mahalanobis distance and a perplexity of 50 is shown in Figure 5.2. As expected the result is similar to [12] even though an improved data generation algorithm was used. It shows that the algorithm has converged and provides a much better separation compared to SNE. The clusters shown have the shape of an aisle

but each of the four aisles is split into two roughly equal parts and a smaller part. This is different from the pathway aisle which is split into two parts, a large and a small part. Although its performance is better than SNE it is still not good enough to solve the use-case as not enough information is transferred from the high to the low dimensional space.

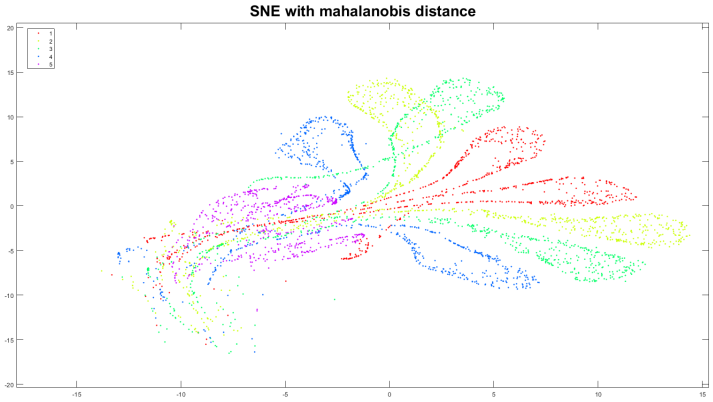


Figure 5.1: Plot showing the data mapped using SNE combined with mahalanobis distance and a perplexity of 50. The colours show the label information.

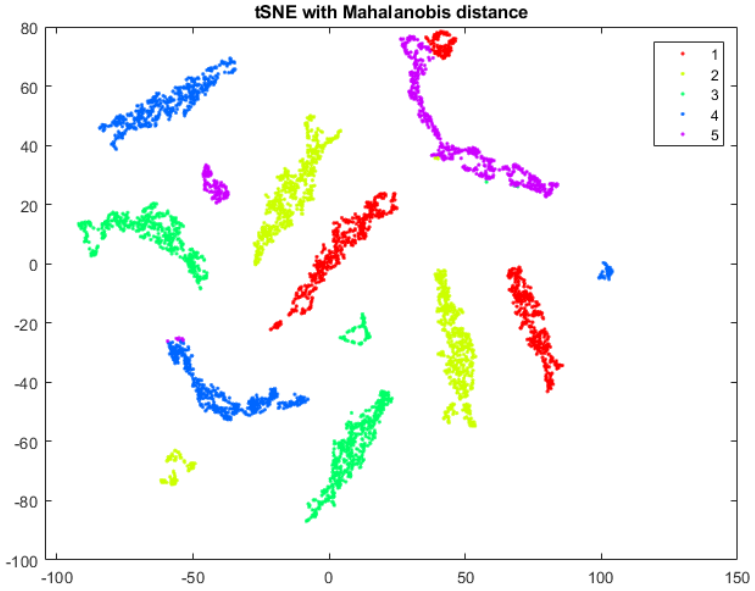


Figure 5.2: Plot showing the data mapped using t-SNE combined with mahalanobis distance and a perplexity of 50. The colours show the label information.

5.1.3 *SOM*

With the traditional dimensionality reduction techniques performing insufficiently the next step was to try SOM which was run as described in Chapter 4. The map size of the self organising map is set to a 12x62 hexagonal grid in order to match the physical size of the warehouse in meters. A training length of 100, an adaptive learning rate starting at 0.5 and ending at 0.05, and a Gaussian neighbourhood relationship were used. The results can be found in Figure 5.3. The pathway aisle, aisle 5, is present almost completely at the bottom which matches the physical map location. Besides this there does not appear to be much structure in the map itself, the other aisles are intertwined in the region above which does not show any clear separation. In its normal state SOM turns out to not be suitable for the use-case which is similar to the conclusion of [17].

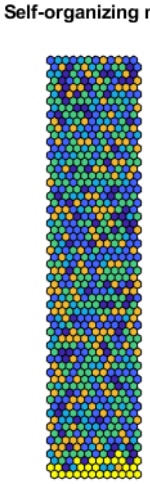


Figure 5.3: Plot showing the data mapped using SOM. Orange is aisle 1, light blue is aisle 2, green is aisle 3, dark blue is aisle 4, and yellow is the connecting aisle 5.

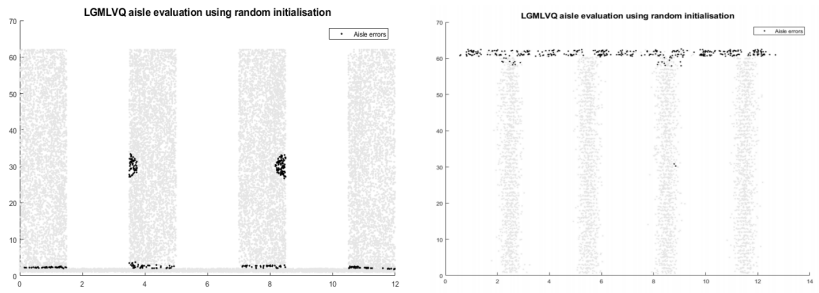
5.2 SUPERVISED METHODS

This section will list and discuss the results of the supervised methods as explained in Chapter 4. The LGMLVQ algorithm will be reviewed based on the classification of the correct aisle and the results will be compared to the results of [12]. The other two supervised methods tested are the novel methods, Local linear t-SNE mappings, and supervised SOM. Local linear t-SNE has been developed by [10] and has not been used for this use-case. Supervised SOM is a novel approach presented in Section 4.2.3. The results of these methods will be compared to the more traditional methods discussed previously and will

be reviewed based on the amount of information transferred from the high dimensional space to the low dimensional space.

5.2.1 *LGMLVQ*

Figure 5.4a shows the evaluation of LGMLVQ which was executed with five prototypes and random initialisation as [12] concluded that random initialisation resulted in the highest accuracy. The black dots indicate a wrong aisle while the greyed out dots are classified correctly. Compared to [12], the results of which are depicted in Figure 5.4b, the results are different which is likely due to a different way of simulating the data. There are two situations in which the algorithm makes a classification mistake. The first situation is related to the points which are the most difficult to classify, the points on the border of an aisle and the pathway aisle. On this location the aisles, and thus the points, are similar to each other so it is a logical classification mistake to make. The fact that it classifies data points at this difficult location wrong might indicate that the location estimation within the aisle is correct as well which has not been tested in this thesis. The other situation in which the classification is wrong is in the middle of the two aisles in the centre. There are two semicircles which are classified wrong. Even though there are two locations where the accuracy drops the algorithm is capable of accomplishing the most important task, determining the correct aisle with a high degree of accuracy and with the more realistic data simulation it performs even better than was shown in [12].



(a) The figure depicts the LGMLVQ aisle evaluation with random initialisation. The black dots indicate a wrong aisle while the greyed out dots are classified correctly.

(b) This figure is created by [12] and shows the same algorithm as Figure 5.4a used on a differently generated data.

Figure 5.4: Comparison between the LGMLVQ performance on the new data (left) and the performance of LGMLVQ as seen in [12] (right).

5.2.2 *Local linear t-SNE Mappings*

This approach is a very interesting approach since it promises to provide an explicit mapping function which makes out-of-sample classification very fast. The results are shown in Figure 5.5 and are obtained using an LGMLVQ model with five prototypes and random initialisation. This is then passed to the non-linear t-SNE mapping function with 300 epochs, an adaptive learning rate starting at 0.8 and ending at 0.1, and a perplexity of 50. The ouput is one of the most interesting results yet. From Figure 5.5 it becomes clear that the inclusion of the LGMLVQ model greatly improves the separation possibilities. All five aisles have an aisle like shape in the lower dimensional space and the relative position of all aisles is correct. This means that the aisles are almost perpendicular with respect to the pathway aisle, the order of the aisles is correct, and they are shaped like an actual aisle. This result combined with the advantages of having an explicit mapping function make it a very interesting option for the use-case at hand.

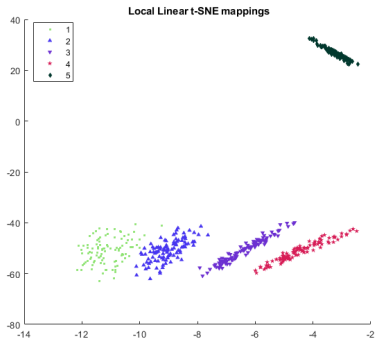


Figure 5.5: The figure depicts the Local linear t-SNE mappings. The black aisle is the pathway aisle number five. The other four groupings are aisles one to four.

5.2.3 *Supervised SOM*

This section discusses the first novel extension of the SOM algorithm presented in Section 4.2.3 which modifies SOM from an unsupervised algorithm to a supervised algorithm. This is done by using the distance metric of LGMLVQ in order to provide better separation and clustering. For the SOM grid again a grid of 12 by 62 is chosen in order to be similar to the actual warehouse shape. The matrices used for the distance calculation come from an LGMLVQ model with five prototypes and was initialised with random initialisation. The results are shown in Figure 5.6. Although there is some minor clustering present the aisle like structures are not present in the lower dimensional space. The middle two aisles, 3 and 4, are often clustered together. Most of the grid

however is represented by empty grid cells and the structures which are there do not seem to contain any useful positional information.

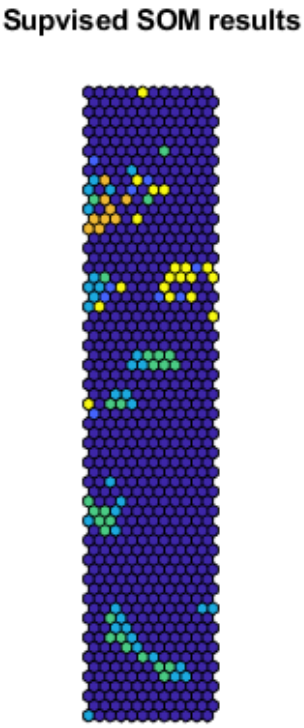
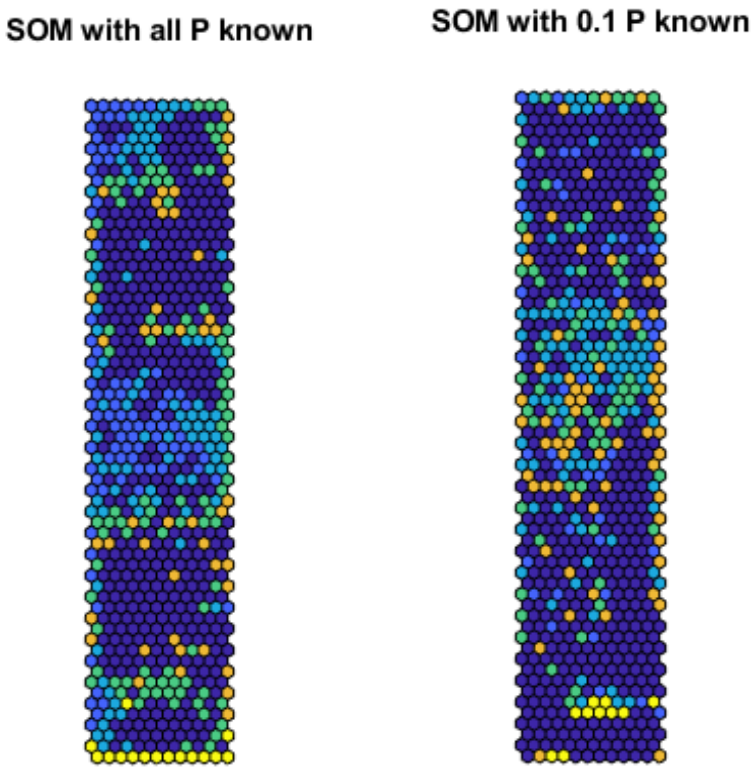


Figure 5.6: This image shows the results of supervised SOM. Orange is aisle 1, light blue is aisle 2, green is aisle 3, darker blue is aisle 4, and yellow is the connecting aisle 5. Empty cells are represented as the darkest shade of blue.

5.3 SEMI-SUPERVISED METHODS

This section discusses the results of the novel semi-supervised version of SOM presented in Section 4.3.1. P is initialised with the true signal strengths based on where the SOM grid neuron is located. Depending on how much supervision you want P can be initialised to a completely filled grid, an empty grid, or somewhere in between. The version with neighbourhood information is used which is listed in Chapter 4 since it was hypothesised that it would perform best. The first test executed was by using a completely filled matrix P in order to see if the results were meeting our expectations. As with the previous SOM tests a hexagonal grid of 12 by 62 was used with the same hyperparameters. The influence of both α and β was set to 0.5 as that appeared to provide the best results. The results of this experiment are shown in Figure 5.7a. The hypothesis was that SOM combined with a small amount of supervised information would drastically improve the

results. This turned out to not be the case as was shown in Figure 5.7a. The pathway aisle is correctly located completely at the bottom but the other four aisles are wrong. Intuitively knowing a true wireless signal strength would pull all points to the correct location which is not what happened. This might be due to incorrect values of α and β which could be either to low in which case the weights w would be moving too slow in the correct direction or it could be too high which results in overshooting the correct location. Since it is a semi-supervised approach it is possible to provide different amounts of supervised information. The result of only knowing 10% of the true signal strengths P is shown in Figure 5.7b. At which location the true signal strength is known was random which resulted in approximately nine entries for every vertical SOM grid lane randomly distributed. The situation is even more chaotic as not many clusters have been formed. The pathway aisle is also not located completely at the bottom but is more clustered than the other aisles. The other aisles do not seem to carry any relevant information. From the above results it could be concluded that semi-supervised SOM does not work however due to time constraints, the number of parameters to tweak, and its execution time, the semi-supervised version was not tested to its fullest. Especially the effect of using different weight values α and β for both the unsupervised and supervised term could be tested more. In its current form however it is not suitable for the use-case.



(a) This image shows the results of semi-supervised SOM with all P known, so for every grid location a true wireless strength. Orange is aisle 1, light blue is aisle 2, green is aisle 3, darker blue is aisle 4, and yellow is the connecting aisle 5. Empty cells are represented as the darkest shade of blue.

(b) This image shows the results of semi-supervised SOM 10% of P known. Orange is aisle 1, light blue is aisle 2, green is aisle 3, darker blue is aisle 4, and yellow is the connecting aisle 5. Empty cells are represented as the darkest shade of blue..

Figure 5.7: Comparison of the results obtained using semi-supervised SOM. On left is a result obtained with all P known while the result on the right is obtained with only 10% of P known.

6 CONCLUSION

To conclude, the goal of this thesis was to evaluate various dimensionality reduction algorithms and methods to carry out the task of indoor localisation for mobile agents. This localisation was to be done using RSS indicator values because of the availability of wireless networks in indoor environments. The accomplishment of this would allow mobile agents to become a more viable solution in the future.

A new method for simulating wireless signal strengths using random paths is presented which resulted in a more realistic data set. Two novel extensions of SOM are presented modifying the algorithm into either a supervised or a semi-supervised algorithm both of which were tested on the newly simulated data. The results of both algorithms were unsatisfactory however due to time constraints the semi-supervised version was not tested to its fullest. A new technique was tested called Local Linear t-SNE which provided very promising results. The ability to classify out-of-sample points quickly using this approach combined with the dimensionality reduction mapping provided is one of the best results obtained so far. LGMLVQ was tested for reproducibility and combined with the new method for data generation provided a very high accuracy for aisle classification.

6.1 FUTURE WORK

There are various new additions and research directions which can be considered for future research. They have been separated in refinements, which improve methods already carried out, as well as extensions, which consider new possibilities and research directions.

6.1.1 *Refinements*

As stated in Chapter 5 the semi-supervised version of SOM was not tested as extensively as desired. This was mainly due to the large amount of parameters which are available. Especially the effect of the factors determining the influence of the supervised and unsupervised term could be investigated further. In addition to this only the version with the neighbourhood influence incorporated was tested since it was hypothesised that this version would provide the best results. It could however be that the results of the version without the neighbourhood influence are better.

The second refinement would be to test the accuracy of LGMLVQ with more exact localisation within the aisle. This has been tested by

[12] who designed hard errors for wrong aisle classification and soft errors for correct aisle classification but wrong in-aisle classification. The results were promising however due to a different method used for data generation and time constraints this was not feasible for this thesis and left out of the scope. Because the results of LGMLVQ were better on the newly simulated data it might be interesting to perform a similar test on the new data.

6.1.2 *Extensions*

There are three ways to extend the project, one of which was shortly discussed in this thesis. This is the incorporation of magnetic field information into the algorithms. Due to difficulties during the gathering of this data and time constraints this was not done in this project. The information on its own is not enough to perform localisation however the sample collected looks promising enough to be able to improve the existing results. Various aisles have different strengths and within the aisle there are some minor differences as well. The incorporation of this new information can be considered an additional dimension with alternative characteristics compared to the wireless network information.

The second possible extension is not yet discussed in this thesis but is an interesting direction nonetheless. The non-linear t-SNE approach was one of the best performing algorithms yet. The problem is however that you cannot put constraints on what the low dimensional data should look like even though you know what the general result should be. Something which is possible with SOM. The general dimension reduction framework discussed in [10] could be applied to SOM. This however has not been done as of yet. It was attempted during this thesis but due to difficulties concerning the derivatives this part was dropped. It remains an interesting yet difficult research direction especially since the t-SNE version performed very well.

The last suggested extension for this thesis is the usage of time-series information. [24] show a method where initial location hypotheses are determined using a particle filter approach after which the possible locations are narrowed down using the trajectory information. This could improve the results of the mentioned algorithms since the locations which are incorrectly classified are surrounded by places which are correctly classified and the only way to move from one aisle to the other aisle is by using the pathway aisle. The new way of data generation using random paths allows for this but due to the time constraints and complexity it was left out of the scope of this thesis.

APPENDIX

A

Feature Nr.	Mean	Median	Standard deviation
1	0.5043	0.4702	0.1639
2	0.5125	0.4757	0.1697
3	0.5128	0.4732	0.1720
4	0.5198	0.4781	0.1762
5	0.5182	0.4750	0.1768
6	0.5236	0.4795	0.1795
7	0.5203	0.4757	0.1786
8	0.5240	0.4795	0.1800
9	0.5189	0.4751	0.1778
10	0.5208	0.4781	0.1777
11	0.5140	0.4731	0.1737
12	0.5140	0.4753	0.1717
13	0.5058	0.4699	0.1659
14	0.5642	0.5470	0.1155
15	0.5686	0.5504	0.1193
16	0.5733	0.5533	0.1211
17	0.5761	0.5556	0.1237
18	0.5792	0.5573	0.1244
19	0.5801	0.5583	0.1258
20	0.5812	0.5587	0.1254
21	0.5801	0.5580	0.1259
22	0.5793	0.5571	0.1245
23	0.5762	0.5553	0.1238
24	0.5735	0.5531	0.1212
25	0.5687	0.5501	0.1193
26	0.5644	0.5467	0.1156
27	0.4686	0.4290	0.1536
28	0.4751	0.4332	0.1582
29	0.4747	0.4310	0.1603
30	0.4803	0.4349	0.1636
31	0.4785	0.4323	0.1643

Table A.1 continued from previous page

Feature Nr.	Mean	Median	Standard deviation
32	0.4829	0.4358	0.1663
33	0.4798	0.4327	0.1655
34	0.4828	0.4357	0.1661
35	0.4784	0.4323	0.1640
36	0.4801	0.4350	0.1632
37	0.4745	0.4311	0.1598
38	0.4748	0.4333	0.1576
39	0.4683	0.4290	0.1530

Table A.1: The mean, median, and standard deviation values for the magnetic field intensity (μT) for the individual features in the dataset.

BIBLIOGRAPHY

[1] Markus Achtelik, Abraham Bachrach, Ruijie He, Samuel Prentice, and Nicholas Roy. "Stereo vision and laser odometry for autonomous helicopters in GPS-denied indoor environments." In: *Unmanned Systems Technology XI*. Vol. 7332. International Society for Optics and Photonics. 2009, p. 733219.

[2] SungHwan Ahn, Jinwoo Choi, Nakju Lett Doh, and Wan Kyun Chung. "A practical approach for EKF-SLAM in an indoor environment: fusing ultrasonic sensors and stereo camera." In: *Autonomous robots* 24.3 (2008), pp. 315–335.

[3] *Android Pie's throttling of WiFi scanning is crippling some network tools*. May 2019. URL: https://www.xda-developers.com/android-pie-throttling-wi-fi-scans-crippling-apps/.

[4] Josep Aulinas, Yvan Petillot, Joaquim Salvi, and Xavier Llado. "The SLAM problem: a survey." In: vol. 184. Jan. 2008, pp. 363–371. DOI: 10.3233/978-1-58603-925-7-363.

[5] Paramvir Bahl, Venkata N Padmanabhan, Victor Bahl, and Venkat Padmanabhan. "RADAR: An in-building RF-based user location and tracking system." In: (2000). DOI: 10.1109/infcom.2000.832252.

[6] Tim Bailey and Hugh Durrant-Whyte. "Simultaneous localization and mapping (SLAM): Part II." In: *IEEE robotics & automation magazine* 13.3 (2006), pp. 108–117. DOI: 10.1109/mra.2006.1678144.

[7] Marius Beul, Nicola Krombach, Matthias Nieuwenhuisen, David Droeschel, and Sven Behnke. "Autonomous navigation in a warehouse with a cognitive micro aerial vehicle." In: *Robot Operating System (ROS)*. Springer, 2017, pp. 487–524.

[8] Cooper Bills, Joyce Chen, and Ashutosh Saxena. "Autonomous MAV flight in indoor environments using single image perspective cues." In: *2011 IEEE International Conference on Robotics and Automation*. IEEE. 2011, pp. 5776–5783.

[9] Christopher M Bishop, Markus Svensén, and Christopher KI Williams. "GTM: The generative topographic mapping." In: *Neural computation* 10.1 (1998), pp. 215–234.

[10] Kerstin Bunte, Michael Biehl, and Barbara Hammer. "A general framework for dimensionality-reducing data visualization mapping." In: *Neural Computation* 24.3 (2012), pp. 771–804.

[11] Robert M Bwana. "Indoor localisation of a mobile agent using prototype based techniques." In: (July 2018). URL: http://fse.studenttheses.ub.rug.nl/18172/.

[12] Robert M Bwana. "Metric learning and non-linear dimensionality reduction for indoor localisation of mobile agents." In: (Dec. 2018). URL: http://fse.studenttheses.ub.rug.nl/18926/.

[13] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age." In: *IEEE Transactions on robotics* 32.6 (2016), pp. 1309–1332. DOI: 10.1109/TRO.2016.2624754.

[14] Bong-Su Cho, Woo-Jin Seo, Woo-sung Moon, and Kwang-Ryul Baek. "Positioning of a mobile robot based on odometry and a new ultrasonic LPS." In: *International Journal of Control, Automation and Systems* 11.2 (2013), pp. 333–345. DOI: 10.1007/s12555-012-0045-x.

[15] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. "MonoSLAM: Real-time single camera SLAM." In: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 6 (2007), pp. 1052–1067.

[16] MBM De Koster. "Recent developments in warehousing." In: *Libro de ponencias 3rd European Logistic Association Logistics Educators Conference. Lisboa*. 1998.

[17] Ankita Dewan. "Indoor Drone Navigation with t-SNE." In: (June 2018). URL: http://fse.studenttheses.ub.rug.nl/id/eprint/17348.

[18] MWM Gamini Dissanayake, Paul Newman, Steve Clark, Hugh F Durrant-Whyte, and Michael Csorba. "A solution to the simultaneous localization and map building (SLAM) problem." In: *IEEE Transactions on robotics and automation* 17.3 (2001), pp. 229–241.

[19] Hugh Durrant-Whyte and Tim Bailey. "Simultaneous localization and mapping: part I." In: *IEEE robotics & automation magazine* 13.2 (2006), pp. 99–110.

[20] Jakob Engel, Thomas Schöps, and Daniel Cremers. "LSD-SLAM: Large-scale direct monocular SLAM." In: *European conference on computer vision*. Springer. 2014, pp. 834–849.

[21] Nynke Faber, René (Marinus) BM de Koster, and Steef L van de VELDE. "Linking warehouse complexity to warehouse planning and control structure: an exploratory study of the use of warehouse management information systems." In: *International Journal of Physical Distribution & Logistics Management* 32.5 (2002), pp. 381–395.

[22] Ramsey Faragher and Robert Harle. "An analysis of the accuracy of bluetooth low energy for indoor positioning applications." In: *Proceedings of the 27th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2014)*. Vol. 812. 2014, pp. 201–210.

[23] Carlos Galván-Tejada, Juan García-Vázquez, and Ramon Brena. "Magnetic field feature extraction and selection for indoor location estimation." In: *Sensors* 14.6 (2014), pp. 11001–11015.

[24] Andrea Gasparri, Stefano Panzieri, Federica Pascucci, and Giovanni Ulivi. "A hybrid active global localisation algorithm for mobile robots." In: *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE. 2007, pp. 3148–3153.

[25] John A Hartigan and Manchek A Wong. "Algorithm AS 136: A k-means clustering algorithm." In: *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 28.1 (1979), pp. 100–108.

[26] Geoffrey E Hinton and Sam T Roweis. "Stochastic neighbor embedding." In: *Advances in neural information processing systems*. 2003, pp. 857–864.

[27] Han-Sol Kim, Woojin Seo, and Kwang-Ryul Baek. "Indoor positioning system using magnetic field map navigation and an encoder system." In: *Sensors* 17.3 (2017), p. 651.

[28] Dieter Kirchner. "Two-way time transfer via communication satellites." In: *Proceedings of the IEEE* 79.7 (1991), pp. 983–990.

[29] Georg Klein and David Murray. "Parallel tracking and mapping for small AR workspaces." In: *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*. IEEE Computer Society. 2007, pp. 1–10.

[30] Teuvo Kohonen. "The self-organizing map." In: *Proceedings of the IEEE* 78.9 (1990), pp. 1464–1480.

[31] Teuvo Kohonen, Jussi Hynninen, Jari Kangas, Jorma Laaksonen, and Kari Torkkola. *LVQ PAK: The learning vector quantization program package*. Tech. rep. Technical report, Laboratory of computer and Information Science . . ., 1996.

[32] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. "Spotfi: Decimeter level localization using wifi." In: *ACM SIGCOMM computer communication review*. Vol. 45. 4. ACM. 2015, pp. 269–282. DOI: 10.1145/2829988.2787487.

[33] Swarun Kumar, Ezzeldin Hamed, Dina Katabi, and Li Erran Li. "LTE radio analytics made easy and accessible." In: *ACM SIGCOMM Computer Communication Review*. Vol. 44. 4. ACM. 2014, pp. 211–222. DOI: 10.1145/2645884.2645891.

[34] Steven Lanzisera, David Zats, and Kristofer SJ Pister. "Radio frequency time-of-flight distance measurement for low-cost wireless sensor localization." In: *IEEE Sensors Journal* 11.3 (2011), pp. 837–845. DOI: 10.1109/jsen.2010.2072496.

[35] Rongbing Li, Jianye Liu, Ling Zhang, and Yijun Hang. "LIDAR/MEMS IMU integrated navigation (SLAM) method for a small UAV in indoor environments." In: *2014 DGON Inertial Sensors and Systems (ISS)*. IEEE. 2014, pp. 1–15.

[36] Tong Liu, Paramvir Bahl, and Imrich Chlamtac. "Mobility modeling, location tracking, and trajectory prediction in wireless ATM networks." In: *IEEE Journal on selected areas in communications* 16.6 (1998), pp. 922–936. DOI: 10.1109/49.709453.

[37] Laurens van der Maaten and Geoffrey Hinton. "Visualizing data using t-SNE." In: *Journal of machine learning research* 9.Nov (2008), pp. 2579–2605.

[38] Richard JD Moore, Karthik Dantu, Geoffrey L Barrows, and Radhika Nagpal. "Autonomous MAV guidance with a lightweight omnidirectional vision sensor." In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2014, pp. 3856–3861.

[39] Chapelle Olivier, S Bernhard, and Zien Alexander. "Semi-supervised learning." In: *IEEE Transactions on Neural Networks*. Vol. 20. 3. 2006.

[40] Karl Pearson. "LIII. On lines and planes of closest fit to systems of points in space." In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (1901), pp. 559–572.

[41] Nissanka B Priyantha, Anit Chakraborty, and Hari Balakrishnan. "The cricket location-support system." In: *Proceedings of the 6th annual international conference on Mobile computing and networking*. ACM. 2000, pp. 32–43. DOI: 10.1145/345910.345917.

[42] Atsushi Sato and Keiji Yamada. "Generalized learning vector quantization." In: *Advances in neural information processing systems*. 1996, pp. 423–429.

[43] Konstantin Schauwecker and Andreas Zell. "On-board dual-stereo-vision for the navigation of an autonomous MAV." In: *Journal of Intelligent & Robotic Systems* 74.1-2 (2014), pp. 1–16.

[44] Petra Schneider. "Relevance Matrices in LVQ." In: *Similarity-based Clustering and its Application to Medicine and Biology*. Ed. by Michael Biehl, Barbara Hammer, Michel Verleysen, and Thomas Villmann. Dagstuhl Seminar Proceedings 07131. Dagstuhl, Germany: Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2007. URL: http://drops.dagstuhl.de/opus/volltexte/2007/1133.

[45] Petra Schneider, Michael Biehl, and Barbara Hammer. "Adaptive relevance matrices in learning vector quantization." In: *Neural computation* 21.12 (2009), pp. 3532–3561.

[46] Petra Schneider, Frank-Michael Schleif, Thomas Villmann, and Michael Biehl. "Generalized matrix learning vector quantizer for the analysis of spectral data." In: *ESANN*. 2008, pp. 451–456.

[47] P Series. "Propagation data and prediction methods for the planning of indoor radiocommunication systems and radio local area networks in the frequency range 900 MHz to 100 GHz." In: *Recommendation ITU-R* (2012), pp. 1238–7.

[48] Shaojie Shen, Nathan Michael, and Vijay Kumar. "Autonomous multi-floor indoor navigation with a computationally constrained MAV." In: *2011 IEEE International Conference on Robotics and Automation*. IEEE. 2011, pp. 20–25.

[49] Carlos Oscar Sánchez Sorzano, Javier Vargas, and A Pascual Montano. "A survey of dimensionality reduction techniques." In: *arXiv preprint arXiv:1403.2877* (2014).

[50] Yixin Wang, Qiang Ye, Jie Cheng, and Lei Wang. "RSSI-based bluetooth indoor localization." In: *2015 11th International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*. IEEE. 2015, pp. 165–171. DOI: 10.1109/msn.2015.14.

[51] Andy Ward, Alan Jones, and Andy Hopper. "A new location technique for the active office." In: *IEEE Personal communications* 4.5 (1997), pp. 42–47. DOI: 10.1109/98.626982.

[52] Sebastian Wehkamp. "Indoor drone piloting: A literature overview." In: (Sept. 2018). URL: http://http://fse.studenttheses.ub.rug.nl/18554/.

[53] Jie Xiong and Kyle Jamieson. "Towards fine-grained radio-based indoor location." In: *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications*. ACM. 2012, p. 13. DOI: 10.1145/2162081.2162100.

[54] Shengdong Xu, Dominik Honegger, Marc Pollefeys, and Lionel Heng. "Real-time 3D navigation for autonomous vision-guided MAVs." In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2015, pp. 53–59.

[55] Jie Yang and Yingying Chen. "Indoor localization using improved RSS-based lateration methods." In: *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference*. IEEE. 2009, pp. 1–6. DOI: 10.1109/glocom.2009.5425237.

[56] Moustafa A Youssef, Ashok Agrawala, and A Udaya Shankar. "WLAN location determination via clustering and probability distributions." In: *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003.(PerCom 2003)*. IEEE. 2003, pp. 143–150. DOI: 10.1109/percom.2003.1192736.

[57] Moustafa Youssef and Ashok Agrawala. "The Horus location determination system." In: *Wireless Networks* 14.3 (2008), pp. 357–374. DOI: 10.1007/s11276-006-0725-7.

[58] Xiaojin Jerry Zhu. *Semi-supervised learning literature survey*. Tech. rep. University of Wisconsin-Madison Department of Computer Sciences, 2005.