Tampere University

Saiful Islam

INFRASTRUCTURE-LESS POSITIONING

Localization in GNSS-denied environments

Faculty of Information
Technology and Communication
Sciences
Master's Thesis
August 2019

ABSTRACT

Saiful Islam: Infrastructure-less based positioning
Master of Science Thesis
Tampere University
Information Technology, MSc (Tech)
August 2019
Examiner: Assoc. Prof. Elena-Simona Lohan
Examiner: Dr. Philipp Müller

Research on the uses of navigation and positioning services is considered to be stable and such uses have an exponential growth in outdoor environments. The satellite-based positioning systems provide us an affordable outdoor positioning service and it becomes part of our daily lives. However, all the benefits from the satellite systems may be lost when we enter the indoor spaces or such places where satellite signals are not received due to high attenuations of walls and floor materials. Many users such as rescuers, mine workers, firefighters maneuver have their activities in satellite-denied area and need to work without past information about their surroundings. That is why we need some other positioning services, which can aid us in satellite denied areas. Different technologies can be used for indoor positioning. However, the use of indoor positioning systems may involve extra infrastructure and setup, making the indoor positioning system more complicated and costlier than the satellite-based outdoor technology. An enhanced positioning technology with ubiquitous coverage can address these issues that reduce infrastructure dependence. In this thesis, an infrastructure-less based positioning algorithm is studied. This algorithm relies on the aroma fingerprints of any closed areas. Ion mobility-based electronic noses (eNoses) were used here to obtain aroma fingerprints from different locations. The performance of eNoses in the case of location estimation in an indoor environment has been shown in the research. The data used in this research was collected from seven different locations at Tampere University (TAU) campus. Data in empty and crowded spaces were collected for each location for a total of about 600 s. A supervised machine-learning algorithm was used to process and estimate the probabilistic location. The non-parametric fingerprinting techniques were applied to determine the location from the measurements. The non-parametric system trained with a dataset containing location information of known places called the offline phase. In the online phase, real-time data from the unknown places were recorded and matched with the existing dataset to estimate user location. Five different classifiers were studied in the thesis to predict the location of a user. Using the Scikit-Learn library of Python, a software-based model was developed to evaluate different parameters and outcomes of different classifiers. The popular classifier is k -nearest neighbor (kNN), which correctly predicted about 38 percent of the locations. The impact of different distance metrics and the number of closest neighbors in the localization accuracy were also explored in the thesis. In addition, dimensionality reduction techniques were also applied to reduce correlations between the different electrodes as well as to reduce computational time and complexity. From the results, it is observed that when Principal Component Analysis (PCA) was implemented, the support vector machine anticipated an even more satisfying outcome from the kNN. On the other side, PCA selects eigenvectors that have more variances and removes those with fewer variances that sometimes do not fit with other classifiers. In terms of accuracy, the potential result was achieved from an unusual classifier called Stochastic Gradient Descent (SGD). SGD classifier estimated an object's location correctly up to 53 percent times under certain conditions. Also, some additional experiments were conducted by training the model with different environments. Classifiers are noted to be able to obtain improved accuracy when the model has sufficient environmental data. For example, when the model is well trained, the Random Forest Classifier (RFC) delivers more accurate results than the lightly trained model. On the other hand, RFC delivers poorly accurate results when the model is lightly trained. The impact of the data size on the accuracy was also studied in the thesis. The final experiment demonstrates that most classifiers perform well when the size of the training data is large compared to the size of the test data. However, even with small training data, the kNN classifier with Euclidean distance performed better under all circumstances.

It is hard to state which classifier is the winner from the experiments, but the highest predictive accuracy was achieved by the SGD classifier. SGD output, however, is not stable over time, and with the increase in test size, accuracy decreases. Instead, the stable performance of the kNN classifier with Euclidean distance makes it more reliable to use in any conditions. Also, there were some problems mentioned in the thesis before one could recommend using aroma fingerprints as a trustworthy method of localization.

The originality of this thesis has been checked using the Turnitin Originality Check service.

PREFACE

This thesis has been written for the department of Computing and Electrical Engineering at the Tampere University, Finland.

I would like to express my deepest gratitude to my supervisors, **Associate Professor. Elena Simona Lohan** and **Dr. Philipp Müller** for their continuous support, technical guidance and inspiration during my master's studies at Tampere University. I would also like to thank my Tampere friends and classmates for their help in my studies.

Finally, I would like to thank my parents for their motivation, love and care. Special thanks go to my sister for her hospitality during my studies in Finland.

This thesis is dedicated to my mother **Farida Islam**

Tampere, 26 August 2019

Saiful Islam

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

A	Frequency-dependent path loss component
$\boldsymbol{\beta}$	Mini Batch
$\lvert\boldsymbol{\beta}\rvert$	Eigenvalues of mini batch
c	Light Velocity
d	Distance between the source and target
Δd	Range difference between two-point
$fi(x)$	Loss function
$\nabla f(x)$	Gradient Descent
n	Path loss coefficient
$\boldsymbol{\eta}$	Step size of mini batch
PL	Path-loss
Sw	Intra-class variance
Sb	Inter-class variance
$tarrived$	Time of Arrival
$tsent$	Time of transmission
Δt	The difference in the received time
$\boldsymbol{W}$	Fisher's criterion
$\boldsymbol{xref}$	Known position of a reference point
$\boldsymbol{yref}$	Known position of a reference point

AAS	Adaptive Array System
AI	Artificial Intelligence
BDS	BeiDou Navigation Satellite System
CDMA	Code-Division Multiple Access
CV	Cross-Validation
CWAs	Chemical Warfare Agents
DoA	Direction of Arrival
DoD	Department of Defense
DOP	Dilution of Precision
EKF	Extended Kalman Filter
eNoses	Electronic Noses
FDMA	Frequency Division Multiple Access
GIS	Geographic Information System
GLONASS	Russian Global Navigation System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IMS	Ion Mobility Spectrometry

IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IoT	Internet of Things
IPS	Indoor positioning system
IR	Infrared Radiation
kNN	k-Nearest Neighbor
LBS	Location Based Service
LDA	Linear Discriminant Analysis
LiDAR	Light Detection and Ranging
MEMS	Micro-Electro-Mechanical Systems
ML	Machine Learning
MLE	Maximum Likelihood Estimation
MLR	Maximum Likelihood Ratio
NAVIC	Navigation with Indian Constellation
NLOS	Non-Line of Sight
PCA	Principal Component Analysis
QZSS	Quasi-Zenith Satellite System
RFC	Random Forest Classifier
RFID	Radio Frequency Identification
RSS	Received signal strength
RSSI	Received Signal Strength Indicator
SBS	Switched Beam System
SGD	Stochastic Gradient Descent
SMP	Smallest M-Vertex Polygon
SNR	Signal to Noise Ratio
SVM	Support Vector Machine
TDoA	Time Difference of Arrival
ToA	Time of Arrival
ToF	Time of Flight
UE	User Element
UWB	Ultra-wideband

1. INTRODUCTION

1.1 Motivation and state-of-the-art

Locating and directing is the elementary prerequisite to trace out once way. The information of the present location allows us to discover the best way to move. Determining the own position can figure out by one of the most popular systems by using Global Navigation Satellite System (GNSS). Satellite-based navigation systems have been developed for outdoor use to assist in locating and tracking cars, aircraft, marine vehicles, humans, livestock, and assets. GNSS accuracy is very good in outdoors and especially in the places where satellites are directly visible i.e. having Line of Sight (LoS). There are methods to assist GNSS signal via the cellular networks in the urban area where tall buildings frequently block the LoS. But what happened where the GNSS signal is unable to reach, or a very weak signal? What happened in the indoor area? What happens if we want to locate specific particles/people/nodes in the indoors? Global navigation systems are not suitable for indoor use, where countless applications require precise positioning. Precise and quick indoor localization and tracking has numerous possible use cases such as safety and security, resource productivity, autonomous vehicle navigation, augmented reality, virtual reality, emergency assistance. Different new navigation systems will be discussed in this thesis that can provide a precise position estimation for GNSS denied areas or users. These techniques are called infrastructure less positioning or semi infrastructure less. Self-localization ability is highly desirable for different wireless sensors networks. Detecting data or sensing something is not useful unless the position of the data is acknowledged at the receiving end/receiver. For instance, handling emergency (in a large building, underground, caves, and subway) could be easy if the location of rescue teams could be obtained at any moment. For this purpose, a range of self-positioning techniques has been documented in this thesis. Mobile robots, on the other hand, are gradually taking pains to inspect radioactive materials for multiple applications in a dangerous place (underground or some limited nuclear regions) to replace people and boost operational safety. Thus, indoor positioning is a complicated method. The positioning systems should be robust and accurate. There are numerous technical methods available for designing an indoor positioning system. Some of them provide very high precision (mm level accuracy) at the expense of cost and sometimes difficult to enforce

in some locations. Other methods are cheaper and easier to install, but sometimes their accuracy is questionable. Probable technology and physical characteristics that can be implemented for indoor positioning such as RSSI (Received Signal Strength Indicator) based RFID, Wi-Fi, Bluetooth, ZigBee, Ultra-wideband (UWB). Indoor positioning has also been developed for unusual sensor techniques such as camera and image pattern recognition, IR light sensors (modulating light), acoustic detection, magnetic field detection, odor/smell detection, laser, pressure sensors, proximity sensors, gyroscopes, and accelerometry sensors, etc.

1.2 Objectives

The main objectives of this thesis have been to explore methods that do not use any kind of existing infrastructure and only use user-driven tools (mobile phone or other measuring devices) and the appropriate location scenario (such as a map, fingerprint, etc.) where the position and trajectory of the user can be identified. The aim has also been to know the prospective sensors that can be used for infrastructure-less based localization. Furthermore, the studies aim to explore the different sets of standards with the ability to acknowledge the location of the user and to support them in finding the places of their interest (location-based service). The goal is also to study low-cost low power systems with fast deployment, easily accessible and sustainable. The inertial navigation system (INS) uses the combination of inertial devices to the nonstop estimation of the position of orientation and velocity of moving objects [1], [2]. Earlier, these kinds of technology were only used in aircraft, submarines, and some other military applications. Modern smartphones and tablets are integrated with different MEMS sensors described above [3]. Therefore, smartphones could be the cheapest and widely available device for indoor positioning. Not all smartphones are fully equipped with all the sensors that mentioned above. An electronic nose is used in the experimental part, which is small, light and can be carried comfortably by users and it is also commercially accessible on the market.

1.3 Related Works

There are adequate of current research on object tracking based on different types of sensors discovered during the literature studies. Some works that truly arise from infrastructure-less positioning will be illustrated in this section. Among the other techniques for infrastructure-less positioning, an inertial navigation system is one of them. A group of individuals from Aalto University [3] [55] suggested a purely inertial navigation system using modern smartphones and tablets. The idea is called visual-inertial odometry for

accurate instantaneous positioning, which is based on combining measurements from inertial sensors with visual feature tracing from the video. For example, the idea is a stochastic visual-inertial odometry system that allows a durable indoor navigation system with an Extended Kalman Filter (EKF). This approach is fantastic; however, there are still some issues in a dark environment. It requires keeping the video camera open all the time, which is the issue of smartphones ' energy consumption (battery draining quickly). The same group [56] later worked on pure inertial navigation systems. They suggested an upgraded scheme for their prior job. The new proposed model is based on dual integrating alternated accelerations. Utilizing new techniques, they were able to predict smartphone position, velocity and pose in real time by solving the interpretation with an Extended Kalman Filter (EKF). This approach does not require a video camera. Rather they need a zero velocity of the device after some interval to update the combination of the information receiving from different sensors. Estimation of velocity and position comes from accelerometers and gyroscopes. Therefore, to assess the ultimate location, data from accelerometers and gyroscopes are needed.

Another interesting work that used aroma-based fingerprint to study localization. A team of researchers from Tampere University [39] approaches a model where different rooms can be recognized using fingerprint matching. They used electronic noses (eNoses) that can identify and categorize a wide range of scents. They used ion mobility spectrometry (IMS) in their experiment to identify and sense ionized particles. They took measurements from seven rooms in a different environment and created a fingerprint of those rooms. Finally, they matched unknown real-world data with an existing fingerprint and using the kNN classifier they achieved promising results. However, they talked about some issues with the technique and measurement data. The same data was used in the simulation part of the thesis and attempts to enhance the accuracy and analyze the issues addressed.

There are plenty of studies on positioning based on reflection or multipath. Ultrasound and laser or optical range detectors are the most common technologies used for distance finding in self-directed navigation systems. Both methods are using Time of Flight (ToF) technology. The concept is that ultrasonic converters generate ultra-frequency sound waves and measure the flight time of a sound wave from the transmitter to the target device and back to the receiver. The ToF estimates the round-trip time between the discharge of ultra-frequency sound waves and the return of the pulse-echo arising from its reflection from an object. The measured total time is then divided by two and then multiplied by the speed of light to get the real distance. The same procedure applied with laser radar (LiDAR) for measuring distance.

1.4 Author's Contribution

The thesis focuses primarily on studying a system capable of working without the use of any existing infrastructure. The author's contribution to the thesis is given below

- Literature review of different infrastructure less based positioning algorithms
- Studies to identify sensors used for positioning in GNSS-denied environments
- Studies algorithms of hybridization for an infrastructure-less positioning algorithm
- Read-through and enhance previous studies on indoor localization using aroma fingerprint.
- Implementation of Python-based machine learning model for infrastructure-free localization.
- Performance analysis of different machine learning classifiers and reverse scenario using the Scikit-Learn library

This thesis is containing seven chapters, including the introduction. Chapter two discusses the theoretical background of many current methods for outdoor and indoor as well as infrastructure-less positioning. Chapter three discusses the different parameters of measurement. Chapter four illustrates the ways for computing position using different measurements described in Chapter three. Chapter five demonstrates research methodology, contains the overall scenario of implementation part as well as the description of data, different tools used in the experiments and illustrations of different machine learning classifiers. Chapter six shows the simulation results, some observations, error sources and results in the summary and applications. Chapter seven is the final section that concludes the entire thesis and possible future work.

2. THEORETICAL BACKGROUND

2.1 Outdoor Positioning

Outdoor positioning mainly refers to satellite-based positioning. Satellite-based navigation schemes have become an essential part of such applications wherever mobility is required. Global Navigation Satellite System (GNSS) defines any worldwide system of satellites that convey signal on the ground for positioning purpose. There are mostly four GNSS systems includes American Global Positioning Systems (GPS), Russian GLONASS (Global Orbiting Navigation Satellite System), European's Galileo and Chinese BeiDou (BeiDou-3/ BDS-3). GPS and GLONASS are fully operational and both are under military control. Galileo and BeiDou are still developing and not fully operational [63] [64]. Galileo is totally under civilian control and free for use. It denotes a true public service that assurances the durability of service providing for specific use cases. The performance and accuracy of Galileo are very much competitive with GPS and sometimes better than GPS. There are some other regional satellite systems such as Indian NAVIC and Japanese QZSS however they are not providing worldwide coverage. Among all other GNSS systems, GPS is the most popular as it is the first global navigation system in the world. GPS was designed by the U.S. Department of Defense (DoD) mainly for military uses. Now they are providing two types of service, one is for civilian uses called standard positioning service and another one is precise positioning service for military uses. The Russian Global Navigation System (GLONASS) provides two levels of service: one is standard positioning service for civilian uses with free of charge and another one is precise positioning service that is restricted for military and authorized persons. Among all GNSS, only GLONASS transmits FDMA signals [61] [62]. However new GLONASS-K satellites are transmitting CDMA signal along with the FDMA signal [63]. The working principles of any GNSS satellites are almost the same. The satellites broadcast a very precise timing signal and data message called navigation message that contains their orbital parameters such as ephemeris and almanac. GNSS receivers on the earth received the navigation message, process the message and estimate position velocity and time. Minimum four satellites are needed for estimating three-dimensional position and time. At first, the receiver calculates its distance from each visible satellite and then calculates a three-dimensional position using trilateration or multi-trilateration technique. Good accuracy can be obtained if visible satellites are broadly spaced in the sky. Receiver clocks synchronization with satellite clocks are very crucial. The satellites carry atomic clocks onboard and they keep very precise timing [64]. For instance, one

microsecond in clock error, the receiver might miss-calculate its position hundreds of meters away. In spite of having very good accuracy in outdoor, GNSS signals are affected by some error sources. Satellite signals slow down when they pass through the atmosphere, an error called ionospheric and tropospheric delay. Multipath can affect satellite signals especially in the urban areas having tall buildings. Multipath generates signal replicas, which may mislead the receiver during the acquisition process. Non-Line of Sight (NLOS) propagation due to hindrance can decrease the power of the signals or sometimes make signals unavailable to track. Dilution of Precision (DOP) also plays an important role in GNSS accuracy. DOP value is high when visible satellites are close to each other and the DOP value is low when visible satellites are rationally far apart from each other. High DOP increases positioning error [64].

Even though initial GNSS technology was developed for military service, GNSS having a variety of applications in public, industrial, scientific and some critical applications. Public applications include personal navigation, road transport, railway, aviation, unmanned aerial vehicles, and maritime navigation. GNSS based positioning also use for surveying, mapping, geographic information system (GIS) and enables location-based services such as tourist information, nearby shops, etc.

2.2 Indoor Positioning

Positioning is also required in indoor environments. After the successful development of the GNSS technologies that can deliver acceptable outdoor positioning solutions, challenges have shifted to the zone where GNSS signals are very week or could not pass in: indoor environment, such as inside a tall building, shopping mall, underground tunnels, caves, mines, etc. Indoor positioning system (IPS) has become the hottest topic of researchers and developers in navigation and positioning area [4]. Employing an indoor positioning system may need additional infrastructure that makes the positioning system more difficult than outdoor GNSS technologies [5]. Locating and detecting objects in an indoor environment can be done by using light signals, sound intensity, radio signals, aroma (ion mobility), magnetic fields and other information from sensors collected by mobile phone or any smart devices equipped with modern sensors (Bluetooth, WIFI, infrared, accelerometer, gyroscope, etc.).

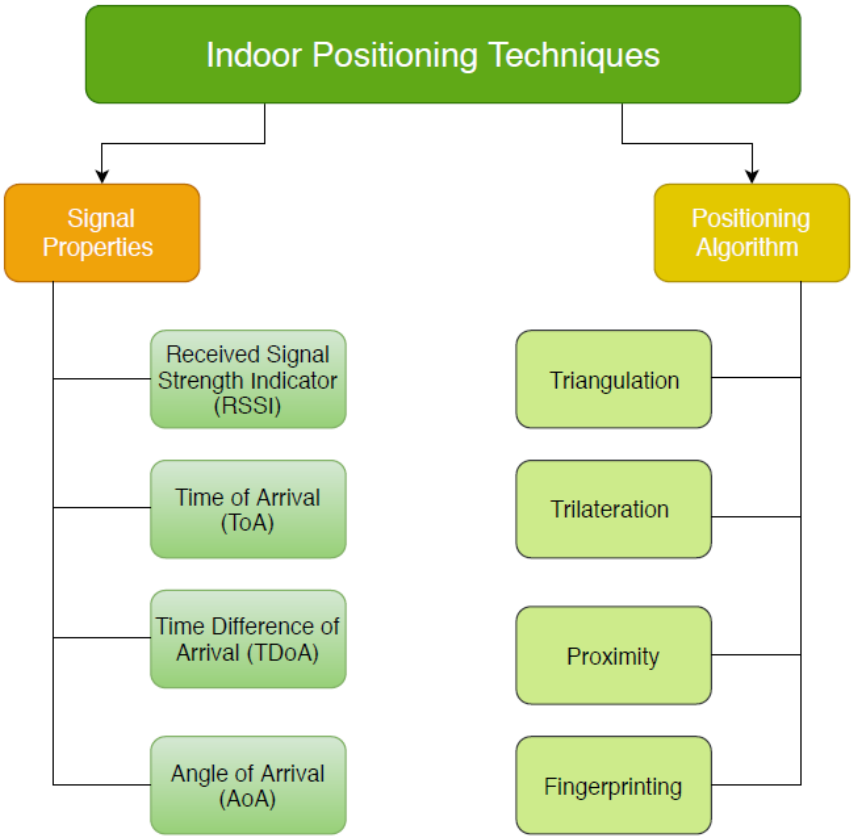


Figure 2.1. Classification of indoor positioning techniques.

Indoor localization has recently seen a growth in attention, because of the possible comprehensive services it can deliver by pulling the Internet of Things (IoT), and pervasive connectivity [6]. In order to provide indoor positioning services, various techniques and technology have been anticipated in the literature. Indoor positioning can be divided into two parts: infrastructure-based positioning and infrastructure-less based positioning. In order to implement infrastructure-based positioning, we need to rely on some preinstalled infrastructures or technologies that could assist to determine the position. Received signal strength (RSS) based technologies (i.e. Wi-Fi, RFID, UWB) are used widely for indoor localization because of easy implementation and can be used by a comprehensive of technologies and devices and relies on existing infrastructure. However, localization based on existing infrastructures does not always guarantee to be available anywhere at any time. There are some localization techniques, which are more flexible to deploy called infrastructure-less based localization.

2.3 Infrastructure-less based positioning

Technologies for positioning habitually relay on wide-ranging infrastructures which confines the coverage of these technologies and make a break down in the user experience once a user attempts to cross the unseen borders of infrastructures. Growth of positioning technologies with universal coverage, which reduces the necessity of infrastructure for positioning [7]. These autonomous systems deliver a more flexible solution than infrastructure-based systems. These systems use sensors such as accelerometers, gyroscopes, magnetometers, ion mobility spectrometers and other sensory information from the positioning device.

With the aim of estimating a person's current positions, continuous recording of sensor data follows the movement of persons. As these positioning techniques do not rely on an exterior infrastructure, hypothetically they can be used anywhere in any environment [8]. The main downsides of infrastructure-less based positioning are error accretions. Sensors can be deceived by environmental noise and therefore misleading of position estimation raises with time and velocity [8].

2.3.1 Magnetic positioning system

There are several methods proposed in the literature to uses magnetic fields for indoor localization. Magnetic-fields map positioning has drawn great attention from the researchers and industry due to its advantage of being functional without any prior infrastructure setup and stable field characteristics. The magnetic-field fingerprinting method develops a map of the magnetic fields that allocated inside a building [9]. Indoor magnetic fields can be distorted by ferromagnetic substantial inside the buildings such as columns, metallic frames and electrical appliances [10]. The possible technique of mapping inside the buildings includes the determination of magnetic flux density [11]. Magnetic flux density is the number of magnetic flux's lines that pass across the surface [12]. If the structure of a building remains constant over a period, the magnetic flux recorded inside the buildings will remain identical during that period [11]. This is the core advantage of the magnetic-field map-based positioning. One shortcoming of this process that magnetic data needs to be recorded in advance before they can deliver accessible positional data. While the magnetic map is ready and plotted on the floor plan, no additional resources are needed to maintain the magnetic field mapping [11].
In order to obtain the position of a mobile device using a magnetic field map and newly measured magnetic data, a probability function can be used. Many likelihood functions

have been suggested in the literature such as Maximum Likelihood Estimation (MLE), Maximum Likelihood Ratio (MLR) and Aggregate Bin Likelihood to estimate positions [13]. Moreover, instead of using one single magnetic map, using multiple magnetic maps have been studied in the literature [14]. A conventional particle system that combines three magnetic maps: horizontal intensity map, vertical intensity map and an orientation information map with an encoded system.
The other methods include the usage of artificially created magnetic fields. In order to create artificial magnetic fields, current carrying coils wire are installed in the specific locations of the buildings. At the receiving end, a mobile sensor records the magnetic field strength from each coil. If the receiver able to measure the magnetic field strength from at least three references coils and if the precise position of these reference coils is known, then the position can be estimated using trilateration techniques [11]

2.3.2 IMU-based positioning system

An interesting technique of infrastructure-less-based positioning arises in the form of inertial navigation [11]. The inertial navigation system (INS) is a self-directed system with decent camouflage that does not rely on any exterior data, nor discharges any energy into the outer space associated with it in the underground, ocean, and skies [15]. Inertial measurement unit (IMU) is the heart of an inertial navigation system (INS). IMUs consist of various accelerometers and gyroscopes that precisely measure linear accelerations and an object's angular velocity concerning a reference point to provide a navigation solution. In general, there are three orthogonal accelerometers and three orthogonal gyroscopes, and a computer required to process and compute position, orientation, velocity, altitude and other information of a moving object. INS is also referred to as “Dead Reckoning” positioning, where navigation starts from a known location and then positioning continues by utilizing the trajectory information (orientation and velocity) compared to a clock [16]. The position estimation accuracy is dependent on the accuracy of trajectory information from the sensors and the time elapsed from the last known location.

The inertial measurement units (IMUs) are divided mainly into two categories. One is a stable platform system and other one is a strap down system. The stable platform system uses a frame (gimbals) where all the inertial sensors are mounted, and the frame is mechanically separated from any external rotational motion [17]. These types of IMUs are normally used in the systems where very precise navigation data are required, for exam-

ple, marine vehicles, underwater vehicles, etc. [18]. In strap down systems, inertial sensors are screwed strictly onto the device, therefore outputs are determined in the body frame instead of the global frame. Strap down systems have summarized mechanical complications thus reducing the physical size and cost of the unit by adding some computational complexity [17].
Inertial navigation systems are used in comprehensive applications together with the navigation of airplanes, tactical weapons, spaceship, submarines, and vessels. One main downside of inertial navigation systems is increasing positioning errors (drift) over time and heat production. Current development of micromachined electromechanical systems (MEMS) inertial sensors are equipped with magnetometers and can reduce drift by up to 5m in every 60 seconds [17].

2.3.3 Localization based on reflection

Localization based on Time of Arrival (ToA) and Time of Flight (ToF) of an optical and wideband signal promises high accuracy indoor positioning thus making it a potential option for a wide range of applications. However, indirect paths (obstacles) are the major challenges of the localization accuracy in multipath propagation. Having Line-of-Sight (LOS) or nonexistence of indirect paths can provide better positioning accuracy. Using wideband signal localization can be achieved in many ways. There are many methods of the localization proposed in the literature. One classic use-case set-up containing a network of nodes located at known places called anchor and some targets in an unknown place that needed to be located. A precise method of estimating the distance between anchor and target is to transmit a short pulse called a range signal from one point (either anchor or target) and measure the time of arrival (ToA) at other points using a mutual reference chronometer. Using the time differences between one point to another point once can easily calculate the distance between the points by multiplying by light velocity. After measuring distances between the points, the location of the target can be estimated using the trilateration technique. A minimum of four distances (3D positioning) from anchors are needed to apply trilateration. Each distance value compels the target to lie on a circle which radius is equal to the distance with an equivalent anchor at the center [19]. By solving the intersection of four circles, the target position can be clearly estimated. Another range-based technique called multi-trilateration may also be used for localization. Alternatively, the range between two points can be obtained from the ratio between transmitted and received signal power as the signal strength decreases exponentially with the distance. Positioning using this method is

called the received signal strength indicator (RSSI) based positioning and Angle of Arrival (AoA) required to measure along with the distances.

Localization accuracy depends heavily on how precise the measured distance between the points is. There are some challenges that influence measurement accuracy such as time synchronization between transmitter and receiver, thermal noise existence, disruptive transmissions, multipath (induced by obstacles), etc.

From the above discussion, it is apparent that most targets have active RF circuitry that allows them to either transmit or receive pulsed signals. This unlocks enormous numbers of applications where smartphones are intended to be incorporated with multiple location-based services (LBS). Alternatively, there are some applications where the target has only a reflector, i.e. the target has no active RF circuit. This is referred to as passive localization.

There is another indoor positioning technique based on a refection called ultra-sound positioning that normally uses Time of Flight (ToF) measurements. Ultrasound transmits a relatively slow reference radio signal to calculate the distance between transmitters and receivers [20]. Variance in ToF between the radio signal and ultrasound signal is measured. The ultrasonic transducer produces a high-frequency sound wave and estimates the sound coming back from the target. Commercial ultrasound range detectors are fitted with temperature sensors and a circuit that adjusts the distance estimation for thermal and atmospheric effects [21] [22] [23].

3. MEASUREMENT TYPES

3.1 Angle of Arrival (AoA)

AoA estimation is the procedure that specifies the direction of arrival of the incoming signal by processing the intruding signal on an antenna array. AoA is also called the direction of arrival (DoA). The AoA estimation can be divided primarily into two parts. Switched beam system (SBS) and Adaptive array system (AAS). SBS uses a specific number of beams to search for the azimuth plane. The angle of arrival is the angle of beams with highest received signal strength. The SBS process is very simple and easy to implement, as it requires only a single receiver and no additional processing techniques are required to measure AoA [24]. In SBS, the complexity of the hardware is very low. However, if the received signal strength is lower than the receiver sensitivity threshold, then SBS will fail to estimate the AoA.

The Adaptive array system (AAS) can direct the beam in any direction by applying weights to the antenna array components. AAS uses N antenna receivers to estimate AoA. Where N is the number of antenna arrays. AAS technique may operate at lower SNR than the SBS. However, more complex hardware is required than SBS. [24]. Figure 3.1 shows that AoA technique uses two known reference points (A, B) and measured angles ($\boldsymbol{\theta_A}, \boldsymbol{\theta_B}$) to estimate the two-dimensional (2D) position of target **X.**

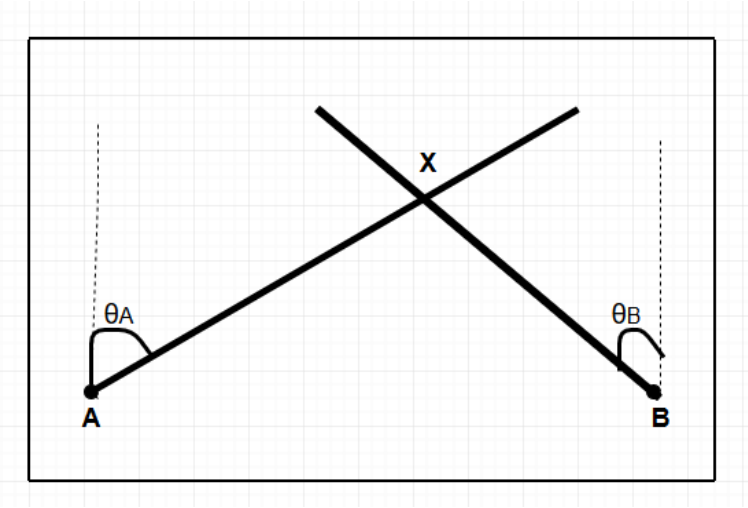


Figure 3.1. AoA estimation (Regenerated from [30]).

There are some benefits to using AoA techniques. The main advantage of AoA is that the target location can be estimated using two measuring units for 2-Dimensional positioning or requires three measuring units for 3-Dimensional positioning. Another plus point is that it does not require (time) synchronization between the measuring units.
There are some drawbacks to using AoA, though. It does not work well in the non-line-of-sight (NLoS) condition, positioning accuracy decreases with respect to multipath increases. In the case of NLoS and multipath, therefore, this technique is not suitable for indoor use.

3.2 Time of Arrival (ToA)

Time of Arrival (ToA) or Time of Flight (ToF) is the simplest and most common ranging techniques for implementing the positioning algorithm. These techniques are based on precise timing information when the signal is transmitted from a transmitter, the precise time when the signal is received at the receiver and the velocity of the signal travels (usually light velocity). Once the timing information is known, the distance between source and destination can be calculated using the equation below [25].

$$d = c * (t_{arrived} - t_{sent}) \quad (3.1)$$

Here c is the velocity of light (3*10^8 m/s)
The calculated distance (d) can be used to determine the position of a target from a reference point. Positioning can be either two-dimensional (2D) or three-dimensional (3D). The circle equation can be as follows in 2D.

$$\boldsymbol{d = \sqrt{(x_{ref} - x)^2 + (y_{ref} - y)^2}} \quad (3.2)$$

Where (x_{ref}, y_{ref}) is the known position of a reference point. Once the distances are calculated from at least three reference points (in 2D) or four reference points (in 3D), it is possible to calculate the accurate position of a receiver using the circle intersection point [25].

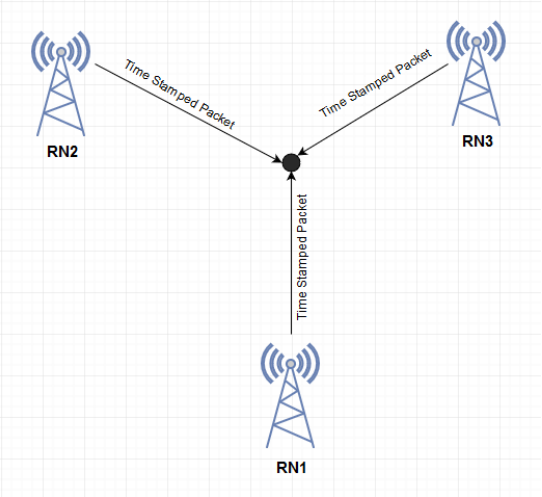


Figure 3.2. Time of Arrival techniques of three Radio Nodes (Regenerated from [26]).

3.3 Time Difference of Arrival (TDoA)

The time difference of Arrival (TDoA) is another most popular ranging method for estimating positioning and sometimes more flexible than the ToA. Although the mechanism is different from ToA, in this case, it does not need to know the exact time when the signal was sent. TDoA exploits the difference in signal transmission times from different transmitters and is measured at the receiving end. When the signal reaches the two reference points, the difference in the received time can be calculated as the difference in the range between the targets and two reference points. Using the following equation, the differences can be calculated [25]

$$\Delta d = c*(\Delta t) \qquad (3.3)$$

Here **c** is the light velocity and $\Delta \boldsymbol{t}$ is the difference in received times at each reference point.In 2D scenario ,following equation can illustrate[25]

$$\Delta d = \sqrt{(x_2 - x)^2 - (y_2 - y)^2} - \sqrt{(x_1 - x)^2 - (y_1)^2} \qquad (3.4)$$

Here $(x_1,y_1)$and (y_1,y_2) are the known position of reference points. For 2D TDoA, at least three transmitters needed to calculate the position of a receiver using the intersections of three or more hyperboloids. The hyperbolic equation can be solved by using a non-linear least square method or Taylor series expansion to linearize the equation[26]. The accuracy of TDoA estimate depends on line-of-sight between transmitter and receiver, signal bandwidth and signal sampling rate at the receiver. Moreover, strict time synchronization between transmitter is also needed [26], unlike the ToF, the timing synchronization is needed between transmitter and receiver.

3.4 Received Signal Strength Indicator (RSSI)

Received signal strength (RSS) based measurement method is one of the simplest and most extensively used positioning technique for indoor [27]. The RSS is defined as the genuine signal power strength received at the receiver, typically measured in (dBm) scale or sometimes (mW) scale. The range between a transmitter (Tx) and a receiver (Rx) can be determined using RSS. The high signal strength obtained means that Tx and Rx are close to each other and the low signal strength obtained means that Tx and Rx are far from each other. The actual distance between Tx and Rx can be determined using different path loss models where the transmission power at the reference power is known. Using the following equation [28] the distance d between Tx and Rx can be estimated.

$$PL = A + n10\log_{10}(d) \quad (3.5)$$

Here A is the frequency dependent component and the received signal strength value is a reference distance from the receiver. The Parameter $\boldsymbol{n}$ is called path-loss exponent or coefficient. The values of n determined how quickly RSS falls with respect to $\boldsymbol{d}$. The value of n determined by the environment, in typical rural cases $n = 2.5$ whereas in dense urban cases $n = 4.5$, however, indoor cases $n = \ 1.6\ to\ 1.8$ [29].

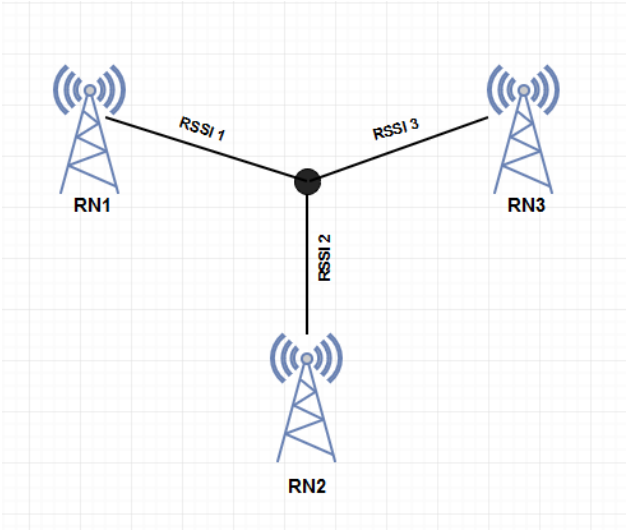


Figure 3.3. RSS based position estimation (Regenerated from [26]).

In RSS-based localization, at least three distances from the reference points to the user devices are required to apply the trigonometric formula to estimate the device location

corresponding to the reference points shown in figure 3.3.RSSI based localization algorithms can be implemented easily and it is cost effective as well as can be used with many technologies. However, it is sensitive to multipath and other environmental noise that can degrade the localization accuracy. After all, several filtering techniques can be used to mitigate multipath impacts and other noises.

Table 3.1. Summary of measurement techniques.

Signal Properties	Measurement Type	Advantage	Disadvantage
Angle of Arrival (AoA)	Angle	Can provide very high positioning accuracy, no fingerprint required.	Required directional antenna and expensive hardware as well as line-of-sight.
Time of Arrival (ToA)	Distance	Can provide high positioning accuracy, no fingerprint required.	Might require time synchronization between transmitters, line-of-sight to ensure high accuracy and possible multiple antenna required at the receiver.
Time Difference of Arrival (TDoA)	Distance	Doesn't require time synchronization among the UEs and radio nodes, fingerprinting not required, high localization accuracy.	An expensive and complex system might require synchronization among the radio nodes as well as high bandwidth needed.
Received Signal strength Indicator (RSSI)	Signal strength	Easy to implement and cost-effective, can be used with various technologies.	Sensitive to multipath effect and other noises from the environment, may require fingerprint in some cases.

4. POSITIONING ALGORITHMS

The positioning algorithm shows how to estimate the position of an object from different measurements [31]. Expressly these algorithms convert different signal properties into either distance or angles and then estimate the real position of an object [32]. For example, once the distance between an object and the reference point is known, the positioning algorithms then determine an object's true position using the estimated signal property [32]. Positioning algorithms, in fact, handle the signal property and estimate a position.

Besides, the accuracy of different algorithms mainly depends on the precision of raw measurements or signal properties [31]. The algorithms have particular pros and cons, therefore using more than one algorithm at the same time can improve the overall positioning accuracy [33]. The most common positioning algorithms are Triangulation, Trilateration, Fingerprinting, Kalman Filter. Etc.

4.1 Triangulation

The triangulation positioning technique uses the geometric properties of triangles to estimate the position of a remote node or a target object [34]. The idea is that using any two reference points, the distance from one reference point to a target object can be calculated if the distance between the reference points and the angle between the reference points and an object is known.

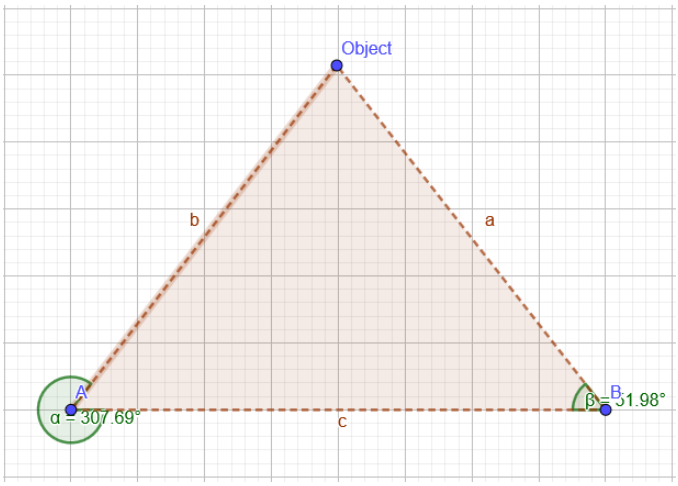


Figure 4.1. Triangulation technique based on angle measurements

In figure 4.1 A and B are two reference points. The distance b can be calculated using sin rule for a triangle. The following equation[34] arranged such a way to find the distance b. However, in order to calculate distance b, angle alpha, beta and distance between two reference points c has to be known.

$$b = \frac{c \sin \beta}{\sin(180-\alpha-\beta)} \quad (4.1)$$

Moreover, if the coverage area is extended along with many reference points, the position estimate may contain some errors that may reduce the accuracy. [31]. Furthermore, hardware necessity for extended area tends to be complex and expensive. Besides, trilateration and triangulation are more often assumed to be the same term, but both terms are obviously described in the thesis.

4.2 Trilateration

Trilateration positioning technique uses geometric properties of circles and sphere in order to estimate the position of a target object. This is one of the oldest and most popular positioning techniques widely used in GNSS as well as indoor positioning. However, this technique is different from the triangulation. In the trilateration technique, at least three distance measurements from the known reference points are used to estimate the two-dimensional position of a target object. Besides, multi-trilateration uses four or more distance measurements in order to estimate position. Using more distance measurements can improved the precision of an object's location as well as altitude [35]. The simple idea is that each measurement from the transmitter expresses a circle of ambiguity where the object can be located. The location of an object can be estimated from the intersection of many circles. The location of an object can be estimated using the Time of Arrival (ToA) measurement at received from the transmitter. Additionally, TDoA measurements can be used to estimate the position. TDoA measures differences in the time the signal arrives from two transmitting stations and determines the hyperbolic positioning of an object. In that case, transmitters need to be synchronized over time. The trilateration positioning technique has been widely studied in [36]. The equation can be derived using the following Pythagorean theorem [57].

$${d_1}^2 = (x_1 - x)^2 + (y_1 - y)^2 \quad (4.2)$$

$${d_2}^2 = (x_2 - x)^2 + (y_2 - y)^2 \quad (4.3)$$

$${d_3}^2 = (x_3 - x)^2 + (y_3 - y)^2 \quad (4.4)$$

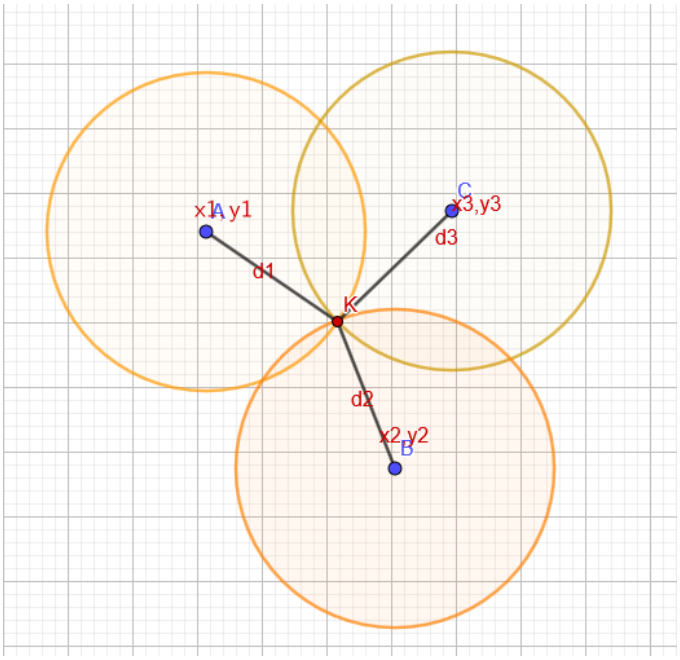


Figure 4.2. Trilateration technique based on timing measurements.

$$x = \frac{AY_{32}+BY_{13}+CY_{21}}{2(x_1Y_{32}+x_2Y_{12}+x_3Y_{21})} \quad (4.5)$$

$$y = \frac{AX_{32}+BX_{13}+CX_{21}}{2(y_1X_{32}+y_2X_{12}+y_3X_{21})} \quad (4.6)$$

Where

$$A = {x_1}^2 + {y_1}^2 - {d_1}^2 \quad (4.7)$$

$$B = {x_2}^2 + {y_2}^2 - {d_2}^2 \quad (4.8)$$

$$C = {x_3}^2 + {y_3}^2 - {d_3}^2 \quad (4.9)$$

4.3 Proximity

The proximity location detection systems examine the position of an object with respect to a known location. However, proximity provides only location information, it does not provide an absolute position of an object [33]. E.g., the beacon is the proximity technology that provides the information about a mobile device somewhere under its proximity but does not tell the actual position where the mobile device is truly located. The systems

depend upon a network of antennas and every antenna has to be in a known position [33]. When a mobile device is identified in motion, the nearest antenna is used to determine its position. When a mobile device is identified by more than one antenna, the antenna with the highest received signal strength is considered to estimate the position [33]. The location of the mobile device is estimated using the received signal strength indicator (RSSI), which is normally used in proximity systems for the purpose of obtaining mobile device position information [37]. Having a mobile device's location information is valuable for initiating different location-based services and applications, such as navigation and tracing [31]. The proximity technique is relatively simple and easy to install in a small area. It can be applied to different types of physical media. Specifically, proximity is frequently used in the system using Radio Frequency Identification (RFID) and Infrared Radiation (IR). Another example of proximity is the cell identification system of the mobile phone. This method depends on the fact that cellular networks can detect the probable position of a user element (UE) by identifying which cell id is using from which base station. The base station and cell location are known by the network, so the handset (UE) is somewhere within the coverage of the detecting cell [33]. However, in order to ensure reliable and wider area coverage, the large spread of readers is required [38]. This large span of readers would make the system complicated and expensive.

4.4 Fingerprinting and Scenes Analysis

Position estimation in the fingerprinting method is performed without the information transmitter distance, angle, and location information. Scene analysis methods indicate an algorithm that collects geographies of a region and then determines the location of an object by matching with earlier recorded fingerprints. Received signal strength (RSS) based fingerprinting mostly used in scene analysis. Moreover, aroma-based fingerprints can also be used for localization [39]. Fingerprints contain information on unique characteristics that distinguishes a scene from another. Scene analysis commonly separated into two phases: Offline stage and online stage (real-time run). In the offline stage, a location survey is conducted in an environment usually within a building. The location grids, labels and corresponding signal strength from nearest anchors or airborne ion mobility are captured. Using the collected data and with the floor plan, a fingerprinting map can be created. During the online stage, a software-based algorithm employs real-time environmental data and earlier collected data from the same location to estimate an object's position [33]. To determine the position of an object, scene analysis commonly

uses some pattern recognition techniques such as k-nearest neighbor (kNN), support vector machine (SVM), neural networks and smallest M-vertex polygon (SMP) and probabilistic methods [33]. The performance and accuracy of the fingerprinting-based localization system are often better than positioning using RSSI [33], [40]. The primary challenge of this method is that the environment can alter rapidly. The received signals can be affected by external noises, multipath, scattering, reflection, etc. which affects the classification accuracy Besides, the offline stage required significant time and hard work to build up the fingerprint map. Therefore, the overall process is time consuming, expensive and complex. Localization based on fingerprinting or scene analysis has been used in RF-based systems such as Wi-Fi, aroma-based systems, for example, monitoring ion mobility and vision-based technologies such as cameras and IR.

5. RESEARCH METHODOLOGY

Different positioning methods have been addressed in the thesis and are mostly divided into two groups: lateration and fingerprinting analysis. Lateration approaches mostly rely on the different signal properties such as Received signal strength (RSS), Time of Arrival (ToA), Angle of Arrival (AoA), Time Difference of Arrival (TDoA) and other signal properties to estimate distance from transmitters to receivers. The received signal characteristics are captured from a particular region in the fingerprinting analysis and then mapped the captured data in a database. In this study, the machine learning algorithm used to process the collected data and train the model to fit with the current environment. After collecting new data set from the same location, the machine learning algorithm can match those data with the existing trained model and predict the probabilistic location of a target object. Usually several classifiers are often used to study characteristics and predictive accuracy of each classifier. Localization using aroma-based fingerprint method is uncommon and that is why this method draws potential attention to the new researchers. Very few pieces of research have been done using electronic noses (eNoses) for localization, according to [39]. Since one of the main aims of this thesis was to study the systems capable of working without the use of any internal infrastructure, which will be cost-effective, sustainable and rapidly deployable. The following block diagram shows a general summary of the entire thesis work.

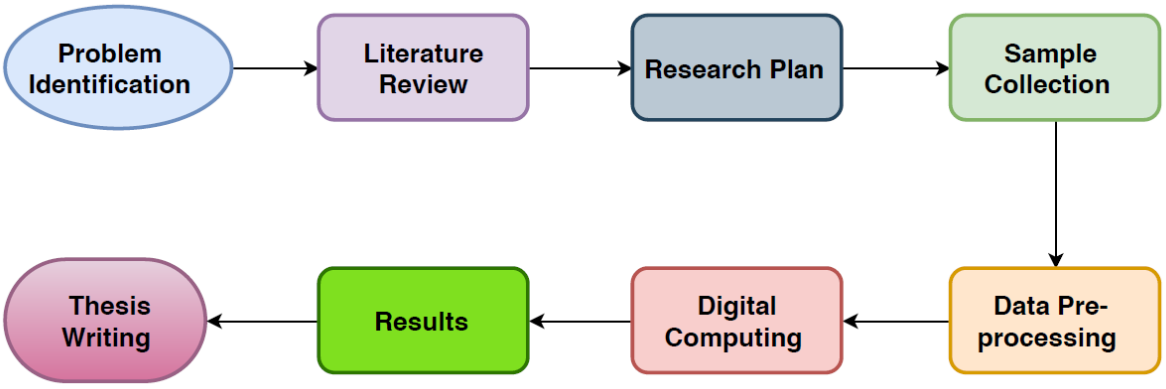


Figure 5.1. Research Methodology

5.1 Work Description

The main implementation part of the thesis is classified into several phases. The data sets for the study were collected from [39]. There were two datasets, taken from the same location with different environmental conditions. Using the given datasets [39] two aroma fingerprint databases have been prepared. A machine learning algorithm was then used to predict an object's location. The idea was to train the ML model using the dataset one with a current situation. The real-world data (second dataset) was then used to predict an object's location. The method was also reversed in the research to analyze how it affects the accuracy of the localization. The reverse method implies that dataset-two was used to train the model and dataset-one used to predict the location. Figure 5.2 shows the work description briefly.

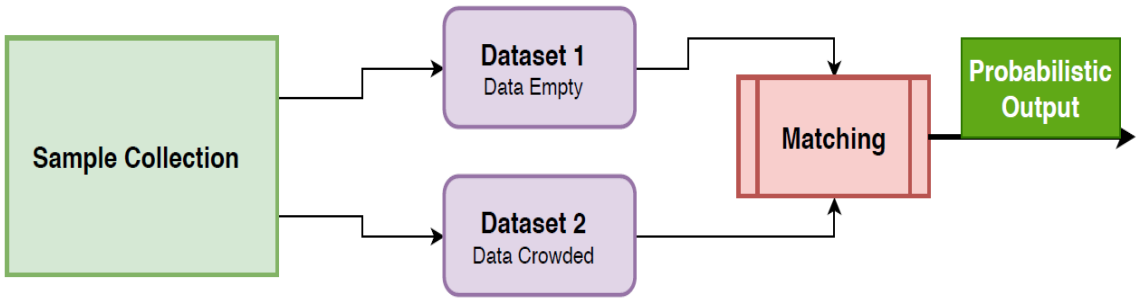


Figure 5.2. Work Scenario

Overall, five classifiers were used to predict the location of an object. Each classifier provides a probabilistic output based on a given dataset. Besides, how the classification accuracy differs with the changing environment and with external noise also studied in the analysis part.

5.2 Data Description

Two datasets from [39] were used in this study: Data Empty and Data Crowded. As indicated by [39] data was collected from seven different locations by a handheld chemical detector (Chempro 100i) at the Tampere University campus. Among the seven locations, there is a small office area (location 1), a coffee room (location 2), an open corridor

(location 5) and four open spaces (location 2, 4, 6, and 7). Besides, locations 6 and 7 are near the food court that is open during the weekdays.

The data sets of about 600 seconds with one-hertz measurement frequency were collected for individual locations. The first data set (Data Empty) was collected during the weekend to guarantee that there were no people in those locations and no food smells. Another dataset (Data Crowded) was collected from the same locations during the weekday when people walked around the locations and cooked foods were displayed and sniffed.

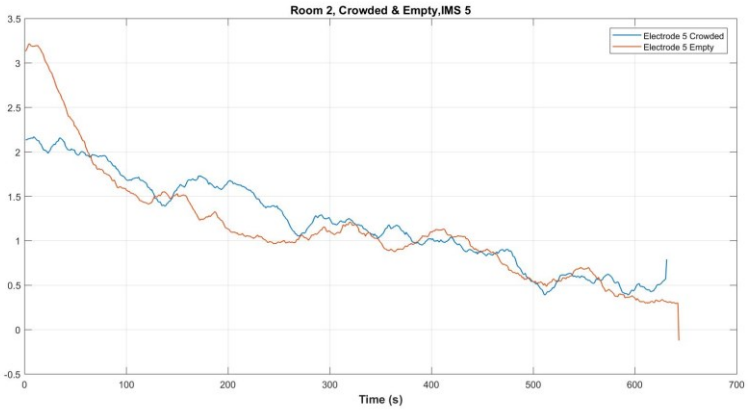


Figure 5.3. Ion mobility plot over time of Room 2.

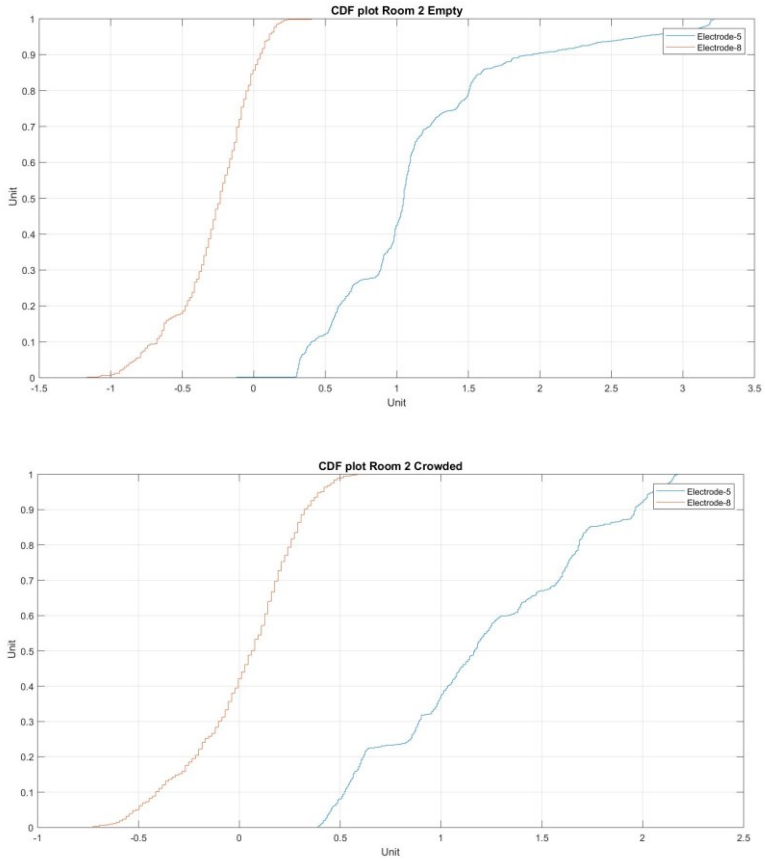


Figure 5.4. CDF plot of Room 2.

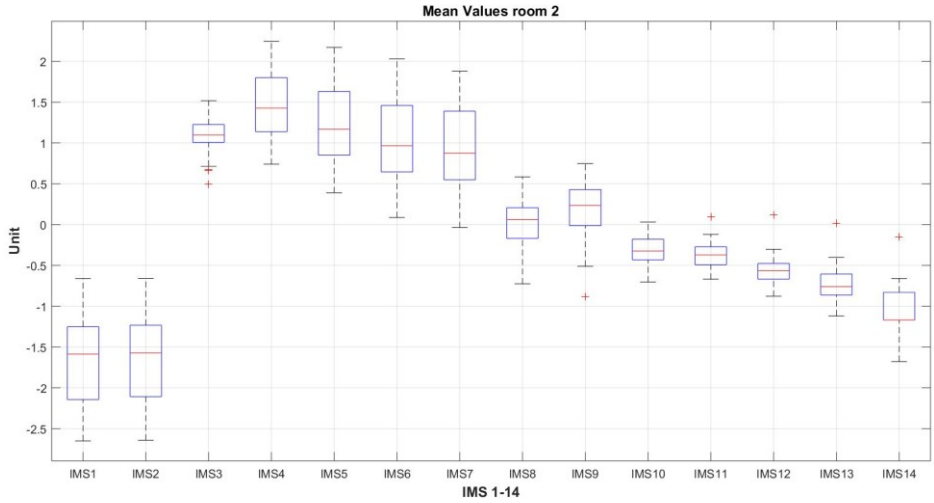


Figure 5.5. Mean value of electrodes 1-14 at Room 2.

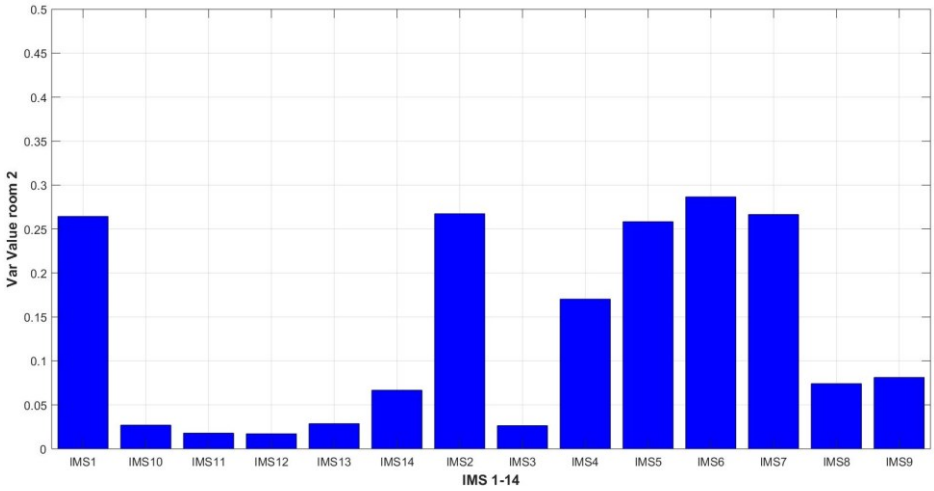


Figure 5.6. Variances plot of room 2 all electrodes.

Figure 5.3 shows the ion mobility plot against time in room 2 with electrode 5 for both empty and crowded situations. It is seen that there is a rapid decrease in the measurement of the electrode's currents at the beginning (about 100s). After certain periods, electrode values are steadily decreasing over time until the end of the readings. It is observed that the values of the electrode decay almost equally under both environmental conditions, which is a fine indication of data reliability. The ML algorithm can easily predict them entirely based on their similarities during the matching phase. An interesting point that can be noticed here is that at the end of the readings there is a sharp decrease in currents due to the operator's body effect.

Figure 5.4 shows the Cumulative Distribution Function (CDF) of room 2 in both empty and crowded situations. The CDF represents the probability of a random variable that is

less than a certain value. The cumulative distribution function of six-sided dies, for example, will look like a staircase. Each step upward will have an additional value of 1/6 plus the previous probability. It will be 100 percent at the end of the graph. The cumulative distribution function is one of the vital statistical instruments and is typically needed by the data scientist to ace the job. The CDF plot of both electrodes in both environments has enough similarity, according to figure 5.4, there are no significant differences observed between the figures. It is also a good representation of data similarities in both environmental circumstances. Figure 5.5 indicates the upper, lower and average values of all electrodes in room 2. Compared to other electrodes, electrodes 1,2,4,5,6,7 have a higher mean value. Figure 5.6 demonstrates the plot of variances of all electrodes in room 2. From the variance figure, it is noticed that the higher variances of these electrodes also result in higher mean values. From the last two figures (Figure 5.5&5.6), the most sensitive and less sensitive electrodes can be easily distinguished.

5.3 Data Collection Tools

A hand-held electronic nose (eNoses) was used to collect measurement data [39]. An eNoses is a device that includes an array of chemical sensing electrodes with a suitable pattern recognition system that able to distinguish and identify various volatile samples [41]. Ion mobility spectrometry (IMS) fitted to detect and analyze different airborne chemicals very quickly. The data sets used in this study were obtained by Chempro 100i (shown in Figure 5.7), which is a handheld chemical detector including IMS. It was originally developed to identify and investigate Chemical Warfare Agents (CWAs) and selected Toxic Industrial Chemicals (TICs) [42]. In this research, the device was used for classifying different airborne. Chempro 100i utilizes several detection techniques that can examine air particles from any location. And depending on their mobility, the device can differentiate and separate ionized molecules [39, 42]. The IMS classifies chemicals using a feature known "mobility" that measures how quickly an ion travels through an electrical field. The mobility associated with size, mass and in order to detach and classify chemicals of interest [43]. The multi-dimensional sensors (16 electrodes) can simultaneously detect a wide range of chemicals. For the data collection purpose, out of 16 electrodes, 14 of them were actively collected data at the same time. Electrodes 8 & 16 were only used to control airflow and therefore were not used for data collection [39].

Figure 5.7. Chempro 100i (source [44]).

The Chempro 100i is an accurate orthogonal sensor and its core remains with the 'Environics' an exclusive ion mobility spectrometry (IMS) sensor [42]. The operation of the device is as comfortable like a mobile phone. However, the device Chempro 100i is a slightly bulky and costly device. However, only its key portion (IMS) is required for localization studies. The latest significant developments of ion mobility spectrometer make IMS chip low-priced, lightweight and widely accessible to the mass consumers. [39, 43].

5.4 Machine Learning Algorithm

Machine Learning (ML) is a technique of data analysis that instructs computers to conduct what actually appears to humans, animals and learn from experiences [45].ML algorithms adaptively improve their knowledge as to the number of learning samples increases. One of the advanced forms of ML is called Deep Learning [45]. In the modern world with the acceleration of big data, ML has become more popular day to day in the field of image recognition, pattern detection, voice recognition, classification, credit scoring, computer vision, etc. The machine learning approach is comprehensive preference while the problem is associated with a very high volume of data and variables and when

there is no convenient formula to solve these issues. ML algorithms are broadly distributed into three parts: supervised learning, unsupervised learning and reinforcement learning. Figure 5.8 presents the different branches of the machine learning algorithm.

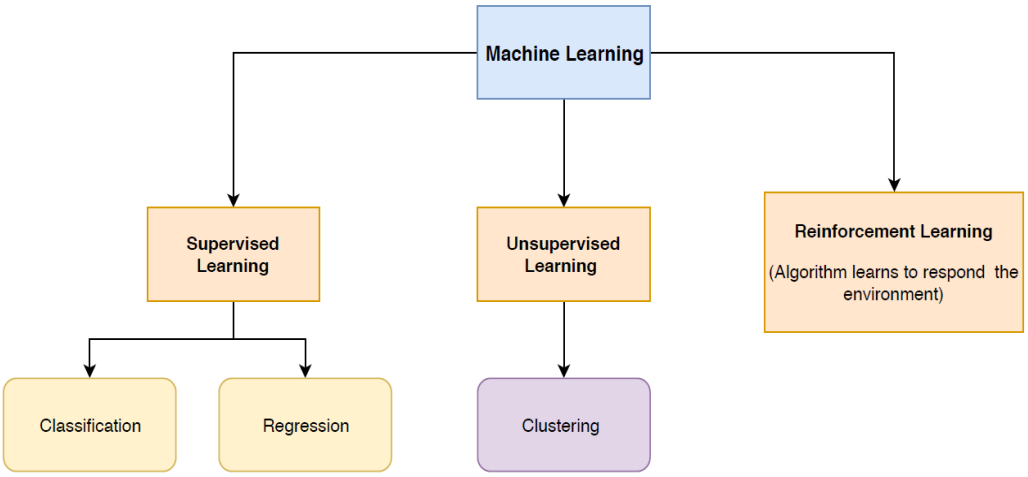


Figure 5.8. Machine Learning Algorithm.

In supervised learning, the machine is trained with a known set of examples: examples include linked inputs and outputs. Then the algorithm indicates a technique of how to reach those input outputs. Once the algorithm finds out the patterns of these input and output data, it trains a model to create reasonable forecasts for the response of new unknown data [45]. A model is ready to create a prediction throughout the learning phase and makes the needed correction when it gives the incorrect estimation. The learning approach continues until the model reaches an expected level of prediction accuracy on training data [46].

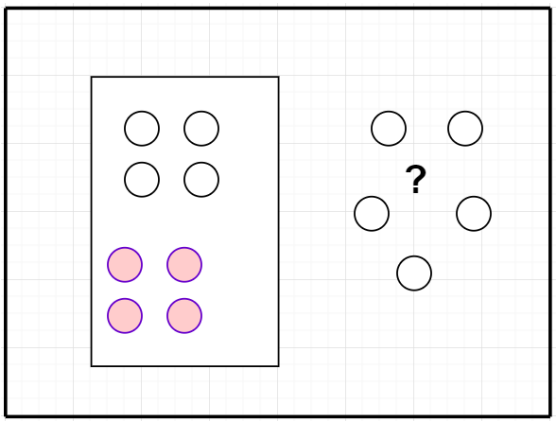


Figure 5.9. Supervised Learning Algorithm.

In order to make predictions of a new dataset, supervised machine learning uses the classification and regression technique. In the classification technique, the model reaches a summary from the current observations about the categories of new observations and where they belong. For instance, an email filtering process model has to predict the email as either “spam or not spam”. Usually, the system looks at training data to get an idea and filter the emails accordingly. However, regression is one of the broadly used statistical approaches, evaluating interrelationship between the variables. There are mainly two types of regression techniques: linear and multilinear. Linear regression utilizes an independent variable to estimate the product of a dependable variable. Multilinear regression uses more than one independent variable to estimate the outcome [58]. There are no predetermined or known labeled data in unsupervised learning. A mathematically developed model systematically reduced redundancy among the data. Additionally, this approach categorized unknown data into several groups using their similarity to each other. Clustering is the popular unsupervised machine learning technique. This approach is mostly applied in order to evaluate the unseen patterns, sequences and classes of the data. Clustering is used to analyze object recognition, market research, etc. [45].

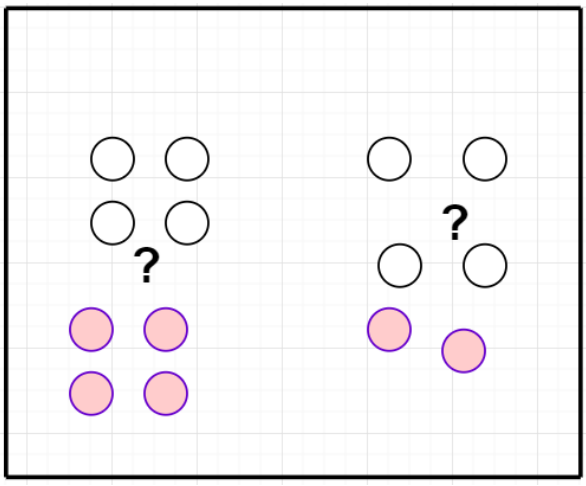


Figure 5.10. Unsupervised Learning Algorithm.

Reinforcement learning is an approach of Artificial intelligence (AI). In this algorithm, any state output depends on the current input and the following input depends on previous output [47]. In the modern world, it is a substantial approach to machine learning where an agent acquires knowledge of how to respond in a situation by conducting activities and observing findings. Depending on the performance, an agent can receive a reward,

accordingly, the agent gets a penalty for the inappropriate performance. Using dynamic programming, an agent decides over time to maximize the rewards and minimize the penalty.

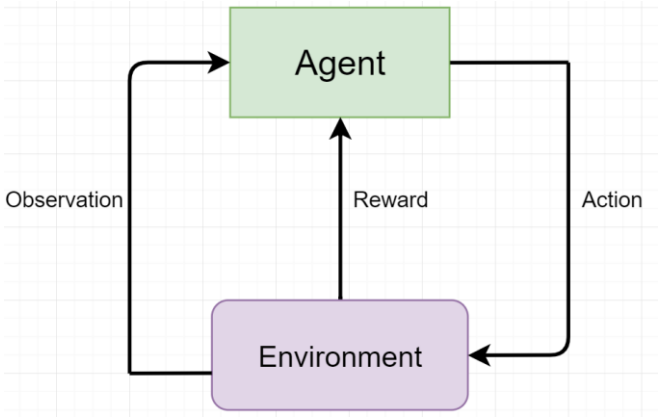


Figure 5.11. Reinforcement Learning.

A supervised machine learning method applied in the thesis for the location estimation. The task is to label the different locations with respect to a known dataset. Therefore, it is a classification problem. In the research, there are several classifiers studied for localization. Figure 5.12 defines the supervised machine learning algorithm used in the research. There are two phases of this work: the training phase and the prediction phase. In the training phase, the model is trained by a dataset with the known labels. In the predictive phase, a new dataset without labels is provided to the model and compares those data with an existing dataset and finally offers the predictive labels of the new dataset. Different classifiers can be used to perform matching. Each classifier has different predicted labels with different accuracies.

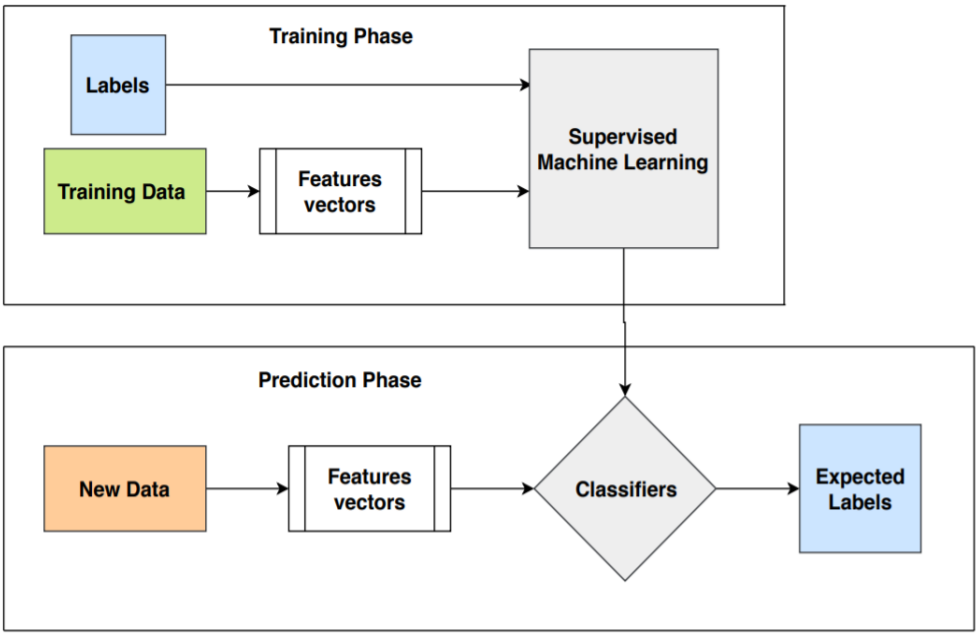


Figure 5.12. An example of Supervised Machine Learning Algorithm.

5.5 Linear Models

5.5.1 Classification

Many machine learning problems can be presented as a classification task. Usually, most of the classification tasks describe as a problem of dividing a vector space into separate areas [48]. Classification problem involves with following factors

$$Samples = \boldsymbol{x}[0], \boldsymbol{x}[1], \ldots, \boldsymbol{x}[N-1] \in \boldsymbol{R^p} \quad (5.1)$$

$$Labels = y[0], y[1], \ldots, y[N-1] \in \{1,2,\ldots,C\} \quad (5.2)$$

$$Classifier = F(\boldsymbol{x}) : \boldsymbol{R^p} \rightarrow \{1,2,\ldots,C\} \quad (5.3)$$

Here, $\boldsymbol{x}$ represents the input, y is the target label and C is the constant. The main task is to figure out the function $\boldsymbol{F}$, which draws the samples perfectly with respect to their labels. For example, study the function $\boldsymbol{F}$ that reduces the number of inaccurate predictions, i.e. the cases $F(x[k]) \neq y[k]$

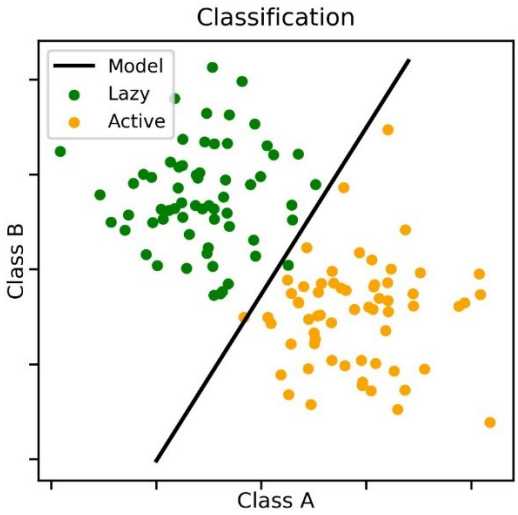


Figure 5.13. Classification Process (Regenerated from [49])

The main goal of classification is to classify samples into particular groups. From figure 5.13, to classify dots based on the training data, the ML algorithm draws a decision boundary that most powerfully categorizes the two classes. A strong decision boundary assigns most of the samples on one side of the boundary that goes to the same class and then most of the samples on the other side that goes to another class.

5.5.2 Regression

Regression is another popular approach to solving problems of supervised machine learning. The output is slightly different from classification tasks; here the output is a real value rather than a category or group [48]. These problems consist of the following elements:

$$Input = x[0], x[1], \ldots, x[N-1] \in \boldsymbol{R^P} \quad (5.3)$$

$$Targets = y[0], y[1], \ldots, y[N-1] \in \boldsymbol{R^P} \quad (5.4)$$

$$Predictor = \boldsymbol{F(x)} : \boldsymbol{R^P} \rightarrow \boldsymbol{R} \quad (5.5)$$

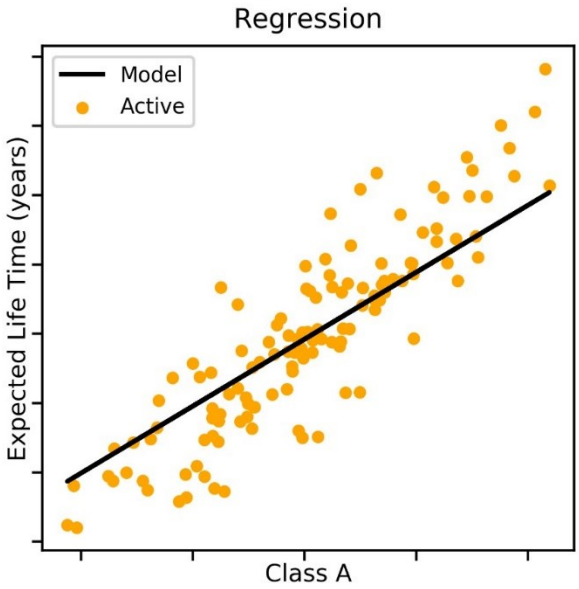


Figure 5.14. Regression Process (Regenerated from [49])

The objective is to get the function $\boldsymbol{F}$ that maps inputs most precisely to their equivalent targets. For example, figure out the function $\boldsymbol{F}$ that reduces the square sum of distances concerning targets and predictors [48].

$$\varepsilon = \sum_{k=0}^{N-1}(y[k] - F(x[k]))^2 \tag{5.6}$$

5.6 Software Overview

Most of the simulation part of this research was performed on the Python platform. Python becomes a popular platform for applied machine learning. It is becoming a fundamental tool for data science. It was designed with a very concise and readable code for the user. Its straightforwardness allows developers to write trustworthy systems. Besides, python attracts many developers as it is simple to comprehend. Furthermore, MATLAB has also used occasionally to generate some figures.

It is complex and time-consuming to apply Artificial Intelligence (AI) and Machine Learning (ML) algorithms. A well-defined structure (frameworks) is necessary to provide the best coding solution. The Python libraries and frameworks are well enriched in order to ensure less complexity and fast programming. Libraries include huge pre-written codes or solutions frequently used in AI and ML. In this research Scikit-learn, Keras, TensorFlow -backend, Pandas, Numpy, etc. are used. Among them, Scikit-learn (Sklearn) is

most commonly used in ML for supervised and unsupervised learning. Different stages of Sklearn API is given below

Initialization: Every model has its individual constructor initiator, for example

$$reg = linear\_model.LinearRegression()$$

Training: Each model implements a $.fit()$ function that trains the model, for example

$$model = reg.fit(X\_train, y\_train)$$

Prediction: There is a prediction stage where model use $.predict\ ()$ function, which estimates probabilistic outputs for new inputs, such as

$$predictions = reg.predict(X\_test)$$

Probabilities: Some models also use a $.predict\_proba\ ()$ function, which provides a class probability for the new input, e.g.

$$probability = reg.predict\_proba(X\_test)$$

5.6.1 Cross Validation (CV)

Performance evaluation of an ML model is very difficult. A dataset is typically split into two sets: a training set and a test set. The training set is used to train the model and the test set is used for testing the model. Evaluating a model with only one dataset is not enough, besides, it remains some drawbacks. The data set splits into the different folds in the cross-validation technique and each fold is used in the test at some point. To perform a CV, In Sklearn library there is a function called $.kFold()$

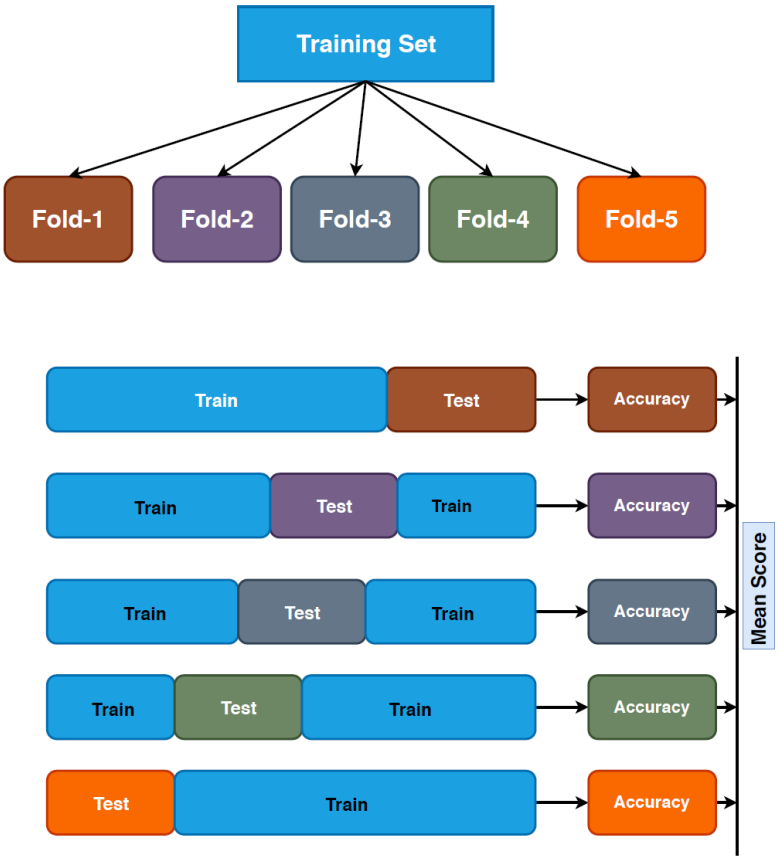


Figure 5.15. Cross-Validation (CV).

In $\boldsymbol{kFold}$ cross-validation (CV), the dataset is divided into $\boldsymbol{k}$ parts where every piece of data is used as a testing set in rotationally at some point. In figure 5.15, the dataset was divided into five pieces to assess cross-validation. In the first cycle, fold-1 is used as the testing set to evaluate the model and the rest of the data is used to train the model. Similarly, in the next cycle, fold-2 is applied as a testing set while the rest of the data is used as the training set. This process is lasting until k times.

5.7 Classifiers

5.7.1 k-Nearest Neighbours

kNN is one of the famous and widely used machine learning and classification algorithm. It is a simple algorithm that caches all possible instances and classifies new instances

based on the measurement of resemblance such as distance [59]. In kNN, the model is structured based on the dataset, therefore, it does not rely on any presumption. In the real world, the majority of cases data do not follow the traditional hypothesis.

Thus, kNN is the best choice for classification tasks when less information available about the distribution of data. In kNN algorithm decision is made based on feature similarity. For each data point, the value of k is the number of nearest neighbors. Usually, k draws the odd integers. For example, when the value of $k = 1$ the algorithm mapped unknown samples based on its nearest neighbors. An object is categorized in kNN algorithm by the majority votes of its neighbors

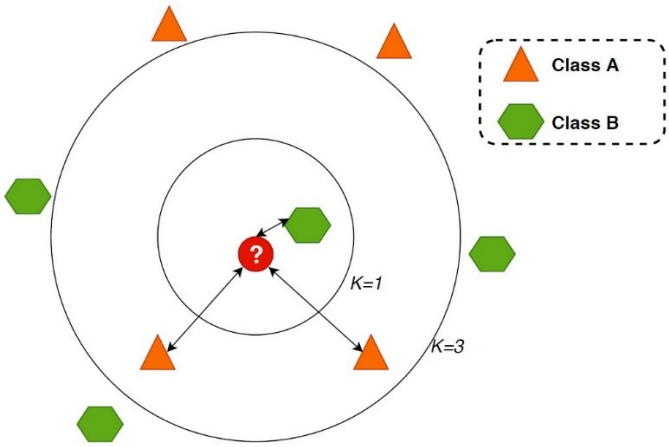


Figure 5.16. kNN algorithm [50].

Figure 5.16 illustrates simplified kNN algorithm. The red circle is the unknown sample, that is to be classified. If the value of $k = 1$, then the new sample is assigned to class B since it is close to a class B sample. If $k = 3$, then the new sample is assigned to class A. Since two orange triangles and one green hexagon are neighbors of a new sample. Therefore, the majority vote goes to class A. kNN algorithm works in the following way [50]

1: Load the data for both training and test

2: Initialized the value of K (Typically k=3, 5)

3: For every point of data

-Calculate Euclidean distance between the new sample and exiting sample

-Sort and store all distances from smallest to highest

-Chose the foremost k points from the stored collection

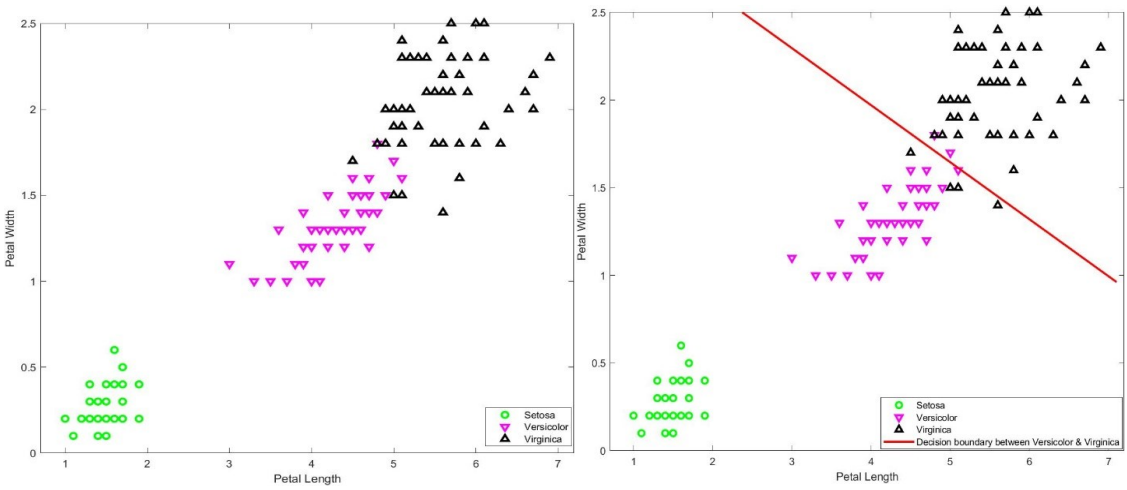
-Labels the k points based on the majority of classes present in the picked points

-Return the mean value for regression and mode value for classification

The main advantage of this model is that it is very simple and easy to implement. Moreover, the algorithm is adaptable in a different environment, for example, it can be used to solve classification and regression problems. The accuracy is relatively excellent and new data can be added or deleted without retraining the model. The value of k is closely related to prediction accuracy. If the value is too small, the algorithm may predict or classify incorrectly, and if the value is too high, then the systems may get slow to iterate so many calculations. Additionally, this algorithm becomes expressively slower as the number of data increases, so it is not the ideal choice for large datasets. However, with enough computer resources kNN can be still convenient to handle issues related to classifying similar objects.

5.7.2 Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) is a dimension reduction technique mainly applied for the pre-processing steps of machine learning and pattern recognition applications. The basic feature of LDA is to reduce dimension by eliminating unnecessary characteristics by transforming them from higher-dimensional space into lower-dimensional space. Besides, LDA discovers the projection that maximized class isolation as far from each other as possible to reduce overfitting and computational expenses [48]. The following figures (5.17) shows the linear classification of Fisher data. Fisher data is an example dataset from MATLAB commonly referred to describe different classifiers.



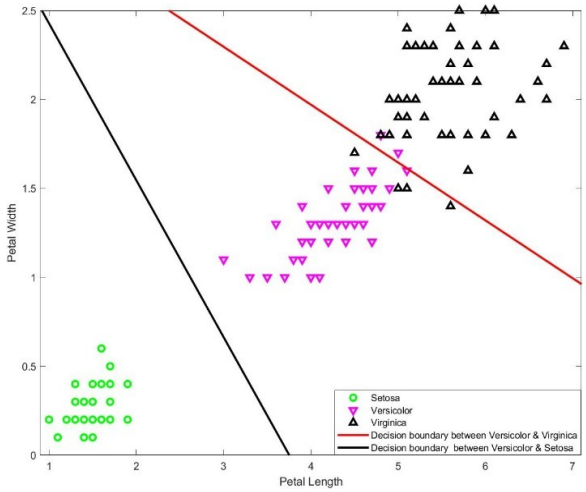


Figure 5.17. Linear classification with Fisher data.

In LDA, the aim is to project the features in upper dimensional space into lower dimensional space. Hence, the job is to be determining a perfect liner projection. It can be achieved in several steps. The first task is to find the distance between the mean of different class i.e. Inter-class variance

$$S_b = \sum_{j=1}^{g} N_j(\overline{x_j} - \overline{x})(\overline{x_j} - \overline{x})^T \quad (5.7)$$

Next step is to calculate intra-class variance

$$S_w = \sum_{j=1}^{g}(N_j - 1)S_j = \sum_{j=1}^{g}\sum_{k=1}^{g}(x_{j,k} - \overline{x})(x_{j,k} - \overline{x})^T \quad (5.8)$$

Then the final step is to determine the projection, which maximizes the inter-class variance and minimizes the intra-class variance. If $\boldsymbol{W}$ be the projection or Fisher's criterion

$$\boldsymbol{W}_{lda} = \frac{Interclass\ variance}{Intraclass\ variance} \quad (5.9)$$

$$\boldsymbol{W}_{lda} = argmax\ \frac{\boldsymbol{W}^T S_b \boldsymbol{W}}{\boldsymbol{W}^T S_W \boldsymbol{W}} \quad (5.10)$$

Using generalized eigenvalue problem solution can be achieved

$$S_b \boldsymbol{W} = \lambda S_{\boldsymbol{w}} \boldsymbol{W} \quad (5.11)$$

Finally, $\boldsymbol{W}_{lda}$ is maximized by choosing λ as the largest eigenvalue.

5.7.3 Support Vector Machine (SVM)

SVM is another supervised machine learning algorithm that is defined by its maximum margin property. In other words, SVM tries to maximize the boundary between the classes. SVM can be used for both regression and classification problems. However, this is most frequently used in the classification task. In high-dimensional spaces, SVM is very efficient. The objective of SVM is to construct a hyperplane in n-dimensional space that clearly categorizes the data point. The vectors that defining the hyperplane are called support vectors.

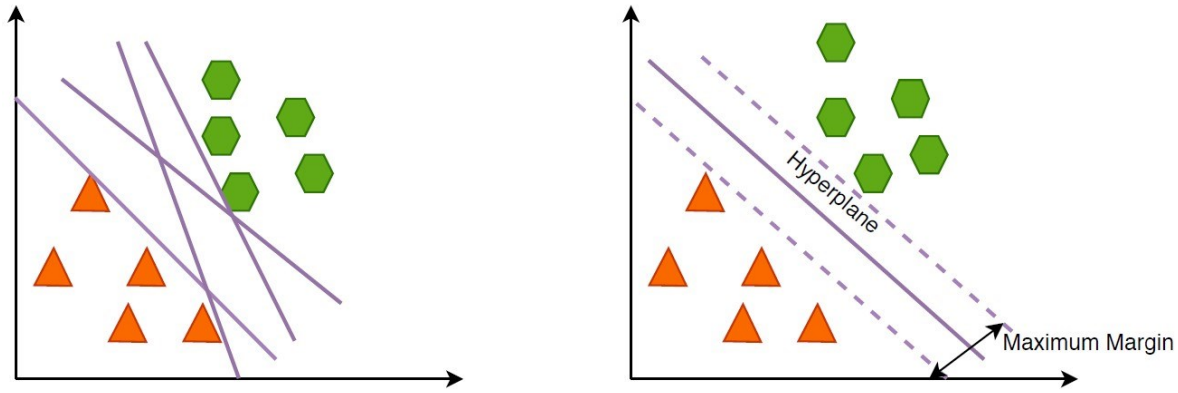


Figure 5.18. Support Vector Machine (SVM) classification techniques [51].

In order to separate two classes of samples, there are many possible hyperplanes. A decision boundary is considered to be a weak boundary if it is passed very close to the samples. Thus, the aim is to find a plane with a maximum margin between the classes. Figure 5.18 (right) shows a clear trend towards the maximum margin. SVM has maximized the margin between two classes with some reinforcement so that future samples are categorized with more confidence. Hyperplanes are the decision margins that help to distinguish the samples. Data samples can be assigned to different classes on each side of the hyperplane. The number of input features determines the dimension of the hyperplane. When the number of input characteristics is three, the hyperplane turns into a two-dimensional plane and the hyperplane resembles a straight line for two input characteristics [51].

The support vector machine becomes a powerful tool when coupled with kernels. The kernel is the useful trick when dealing with multidimensional data. The SVM can be extended to nonlinear margins using the kernel trick. Kernel trick essentially received lower dimension data and converts the data into a higher dimension as well as designs linear SVM there. In Scikit-Learn library, there are many kernels that exist. The RBF kernel is

outstanding among them in the sense that it corresponds to mapping into an infinite-dimensional vector space [48]. Besides, the RBF kernel is generally the best performing one in the mathematical context.

5.7.4 Random Forest Algorithm

Random Forest (RF) is a supervised machine learning algorithm for classification that uses ensemble technique. However, it is also used in the regression problem. RF is a tree-based algorithm that randomly generates several decision trees and supports to fix the problem of overfitting in the decision tree. Decision trees are created randomly using arbitrary features of the given samples. The decision trees are simple to use the if-else statement and make a decision. Random Forest (RF) makes decisions based on the highest number of votes obtained from decision trees.

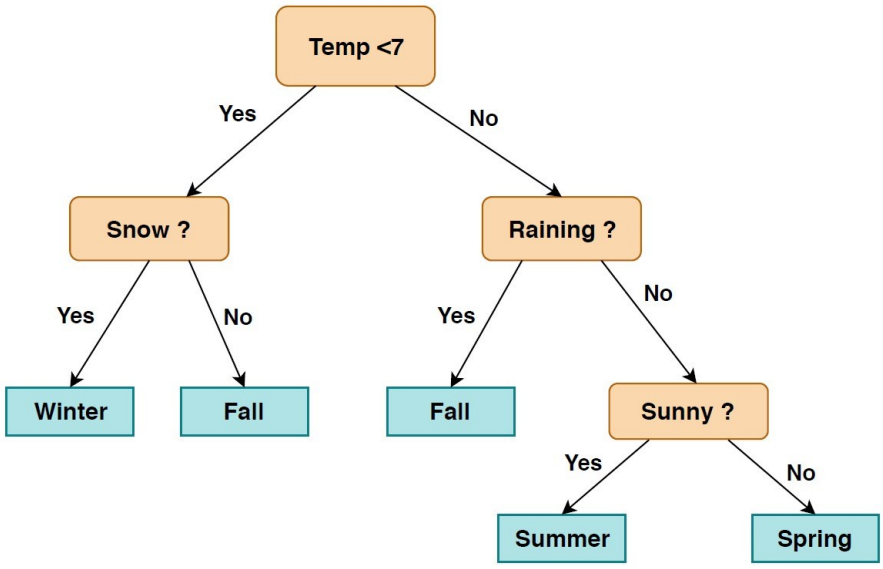


Figure 5.19. Random Forest classifier.

Ensemble technique defines simply a group of decision trees, called together as Random Forest. The accuracy of the ensemble model is better than individual models because it accumulates the results from the individual model and provides a final output. Feature selection is one of the important parts in order to construct decision trees. A technique called bootstrap aggregating (bagging) can be used to randomly pick features from data points. Form the available features in the dataset, a forest of trees is created where numerous training sets are produced with replacement. That implies that at the same time

one feature can be used constantly in separate training sets. The bagging techniques contribute to overcome the problem of overfitting. In the classification problem, RF predicts a class that receives maximum votes from the decision trees. Besides, the average output from the decision trees is considered as the final output in the regression problem.

Random forest algorithm for classification works in the following steps [52]

1. From a given dataset select the random samples
2. For each sample build a decision tree and received predicted output from each tree
3. Accomplish a poll for each prediction
4. Estimate the final prediction based on the majority votes from the trees

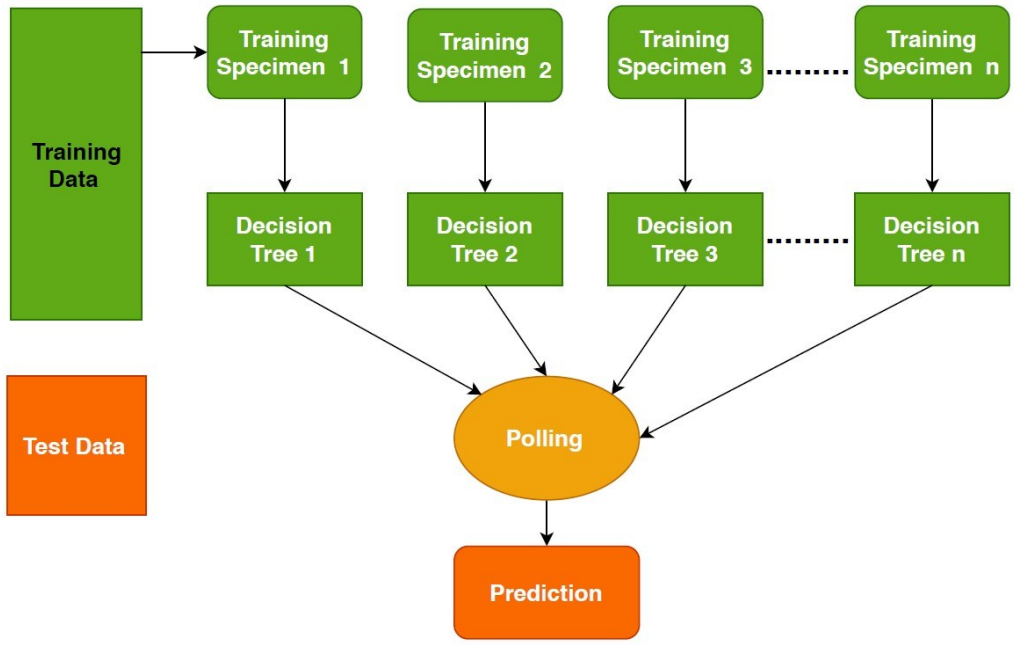


Figure 5.20. Radom Forest Decision trees [52].

Random Forest (RF) algorithm is considered a robust and reliable method for machine learning problems because many decision trees are involved in the whole process. Evaluating the relative status of each feature from the prediction is very simple. For instance, this can be done by a strong Scikit-Learn library. However, dealing with huge random trees makes RF a slow algorithm. In addition, whenever a prediction is made, all trees in the forest must predict the same input and then vote on it and it requires time to finish the entire process.

5.7.5 Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent (SGD) is the perfect selection when dealing with huge dataset. In SGD, a small number of samples are chosen arbitrarily rather than the whole dataset for each iteration. It is helpful to use the entire dataset to ensure better accuracy. However, the problem arises when dealing with the huge dataset.

A stochastic form of the algorithm is therefore very frequently used. While training the deep learning model, it is very often considered that the objective function becomes the summation of a finite number of functions [53]

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) \tag{5.12}$$

Here $f_i(x)$ is the loss function ranked by $\boldsymbol{i}$, based on training data example. In gradient descent, the computational cost per-iteration rises linearly with the size of the training data set (n). Therefore, when training data size is enormous, the computing cost of gradient descent per iteration is very high.

To decrease computing costs, stochastic gradient descent offers a lighter solution. At each iteration, SGD randomly samples i at consistently and computes $\nabla f_i(x)$ rather than computing gradient descent $\nabla f(x)$. The idea is that, stochastic gradient descent utilizes $\nabla f_i(x)$ as an independent estimator of $\nabla f(x)$ [53].

$$E_i \boldsymbol{\nabla} f_i(X) = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\nabla} f_i(X) = \boldsymbol{\nabla} f(X) \tag{5.13}$$

In generalized cases, the SGD algorithm is also called mini-batch SGD. At each iteration, the mini batch $\boldsymbol{\beta}$ which consists of indexes for training data, can be sampled uniformly with substitute. Likewise, we can use

$$\boldsymbol{\nabla} f_{\boldsymbol{\beta}}(\boldsymbol{X}) = \frac{\mathbf{1}}{|\boldsymbol{\beta}|}\sum_{\boldsymbol{i}\in\boldsymbol{\beta}} \boldsymbol{\nabla} f_{\boldsymbol{i}}(\boldsymbol{X}) \tag{5.14}$$

Updating **x** as

$$\boldsymbol{X} := \boldsymbol{X} - \boldsymbol{\eta}\boldsymbol{\nabla} f_{\boldsymbol{\beta}}(\boldsymbol{X}) \tag{5.15}$$

Where $|\boldsymbol{\beta}|$ represents the eigenvalues of mini batch and the $\boldsymbol{\eta}$ represents the step size. Similarly, mini-batch SGD uses $\boldsymbol{\nabla} f_{\boldsymbol{\beta}}(\boldsymbol{X})$ as an impartial estimator for the incline $\boldsymbol{\nabla} f(X)$

$$E_{\beta}\nabla f_{\beta}(x) = \boldsymbol{\nabla} f(X) \tag{5.16}$$

Some other motivations make stochastic gradient descent more attractive than gradient descent. If the training dataset has numerous unnecessary data samples, SGDs can be so close to the real gradient $\boldsymbol{\nabla} f(X)$that a certain number of iterations will be determined appropriate solution to improve the problem [53]. In reality, when the training dataset is huge, SGD only needs a certain number of iterations to determine the relevant solutions while the total computational cost is more economical even for one iteration than that of gradient descent.

6. RESULTS AND ANALYSIS

In this chapter, the performance of the aroma fingerprinting-based localization method and different test results are presented in detail. Additionally, different error sources and limitations of the data and testing methods are exposed. The remaining of this chapter presents a summary of all experiment results.

6.1 Observation

In the simulation part eight experiments have been performed with different parameters. All of the experiments were performed under Python (Python 3.6) platform, using the Scikit-Learn machine learning library. The performance assessment of different classifiers was carried out using the Scikit-Learn toolbox. Different test/training sizes and performance of different distance metrics were also checked within the same classifier. Each classifier is trained and tested with the same training and test data in the same experiment.

Experiment 1

Status: Trained as Crowded, tested as Empty

In the first experiment, the model was trained with Data Crowded and tested with Data Empty. Precisely, the model was trained with crowded data and tested with empty data. The size of the training data is 75 percent of the testing data. In actual instances, less training data is expected. This is, therefore, a good way of studying each other with less training and testing data. Table 6.1 shows the results from experiment 1.

Table 6.1. Results of experiment 1

Classifier	Distance	Classification Accuracy	Value of k
kNN	Euclidean	**37.56%**	5
kNN	Euclidean	37.33%	3
kNN	Minkowski	37.53%	5
kNN	Minkowski	37.21%	3
kNN	Manhattan	36.89%	5
kNN	Cityblock	36.94%	5
kNN	Canberra	29.83%	5
kNN	Cosine	29.97%	5
LDA	--	35.42%	--
SVM	--	34.05%	--
RFC	--	29.55%	--

The kNN classifier with Euclidean distance predicted around 38% location accurately when the value of k is 5. Moreover, in the case of the kNN classifier is it exposed that different values of k have no significant impact on the accuracy of the classifier. However, different distance metrics have some impact on the accuracy of the classifier. Here, the kNN with Cosine and Canberra performed poorly, compared to other distance metrics. Other classifiers like Liner Discriminator Analysis (LDA) and Support Vector Machine (SVM) were well anticipated around 35 percent times. However, Random Forest Classifier (RFC) performance is insufficient compared with other classifiers.

Experiment 2

Status: Trained as Empty, tested as Crowded

Another experiment was conducted by merely reversing the method that attempted to study the results. In this experiment, the model was trained by Data Empty tested by Data Crowded. Similarly, in the second experiment, the size of the training data is 75 percent of the test data. Table 6.2 shows the results of experiment 2.

Table 6.2. Results of experiment 2

Classifier	Distance	Classification Accuracy	Value of k
kNN	Euclidean	29.96%	5
kNN	Euclidean	30.21%	3
kNN	Minkowski	29.57%	5
kNN	Minkowski	29.89%	3
kNN	Manhattan	25.95%	5
kNN	Cityblock	25.94%	5
kNN	Canberra	29.66%	5
kNN	Cosine	**34.12%**	5
LDA	--	31.47%	--
SVM	--	31.31%	--
RFC	--	22.97%	--

In the second experiment, the accuracy of the predicted locations has degraded compared to the first experiment. Similarly, in the second experiment, it is noticed that different values of nearest neighbors (k) have no significant impact on the accuracy of the classifier. However, different distance metrics have some impact on the accuracy of the classifier. In this experiment, maximum accuracy (34%) was achieved by the kNN classifier with Cosine distance when the value of k is 5. Here, Cosine performs much better than in experiment 1 while other distances perform worse. The possible reason for the

overall degradation of performance in the second experiment is that Data Crowded includes more data about the environment than Data Empty. Therefore, greater accuracy would be provided by the model trained with more environmental data.

Experiment 3

PCA Applied

Status: Trained as Crowded, tested as Empty

Principal Component Analysis (PCA) method was used in the third experiment. PCA is a dimensionality-reduction technique most frequently used to reduce the dimensionality of a bulky dataset. It is done by transforming a large set of variables into a shorter one that still holds most of the information in the large set [54]. PCA transformation was used to convert uncorrelated channels from IMS channels. This process reduces the number of channels for classification hence significantly reduces computational time and complexity [39]. The findings of experiment 3 are shown in Table 6.3

Table 6.3. Results of experiment 3

Classifier	Training Size	Classification Accuracy
kNN	75%	29.16%
kNN	25%	32.05%
LDA	75%	28.68%
LDA	25%	28.18%
SVM	75%	**40.83%**
SVM	25%	37.28%
RFC	75%	33.86%
RFC	25%	27.05%

In the third experiment, the results achieved by Support Vector Machine (SVM) were comparatively improved. When the training size is 75 percent, SVM estimates around 41 percent times accurate locations. In addition, the Random Forest Classifier (RFC) also provides a satisfactory accuracy of around 34 percent. It is noticed that there is a clear relationship between training/test size and accuracy. Apart from kNN, with less training data, all the classifiers used in experiment 3 provided less accurate results. On the other hand, it is observed that with less training data, the kNN classifier performed well.

Experiment 4

PCA Applied

Status: Trained as Empty, tested as Crowded

The fourth experiment was carried out simply by reversing the process of the previous experiment. Reducing the number of variables in the dataset sometimes returns at the expense of accuracy. Table 6.4 illustrates the results of experiment 4.

Table 6.4. Results of experiment 4

Classifier	Training Size	Classification Accuracy
kNN	75%	7.9%
kNN	25%	12.06%
LDA	75%	16.04%
LDA	25%	10.83%
SVM	75%	8.16%
SVM	25%	12.70%
RFC	75%	9.53%
RFC	25%	13.23%

In the fourth experiment, inadequate outcomes were obtained by all the classifiers used. The possible reason for the outbreak can be explained using the observation form experiment 2. The model was trained in this experiment with Data Empty, which may have less information about the environment than the Data Crowded. The Principal Component Analysis decreased the number of variables from Data Empty and making them lighter. Consequently, the reverse process (Trained as Empty, tested as Crowded) does not fit well in the fourth experiment for the particular dataset. Although this test also observed the relationship between the training/testing size and accuracy.

Experiment 5

*Stochastic Gradient Descent (**SGD**) Applied*

Status: Trained as Crowded, tested as Empty

An outstanding classifier called Stochastic Gradient Descent (SGD) was used in the fifth experiment. It can be achieved directly from the Scikit-Learn library. Data Crowded was used for better accuracy based on prior experiences. Table 6.5 demonstrates the outcomes of experiment 5.

Table 6.5. Results of experiment 5

Classifier	Training Size	Classification Accuracy	Status	Remarks
SGD	75 %	**53%**	Trained as crowded and test as empty	loss=squared_hinge
SGD	25 %	47%		

Localization accuracy increased to 53 percent in the fifth experiment, which is the highest accuracy obtained from the entire studies. In addition, it is observed that even with small training data, the SGD classifier operates quite well. Besides, several loss functions were accessible in the Scikit-learn library to train the SGD classifier. The square hinge loss function was used in experiment 5 with the SGD classifier. Finally, the SGD classifier with hinge loss, given a maximum probabilistic accuracy of 53 percent. The experimental process is similar to experiment 1 instead of multiple classifiers, only a unique classifier used to focus more on the findings of several experiments.

Experiment 6

Test Size 1-99

Status: Trained as Crowded, tested as Empty

The relationship between training/testing size and predictive accuracy was observed from previous experiments. The final experiment was conducted in order to study more intensely by plotting all the classifiers together with variable testing size from 1% -99%. Figure 6.1 demonstrates the results of experiment 6.

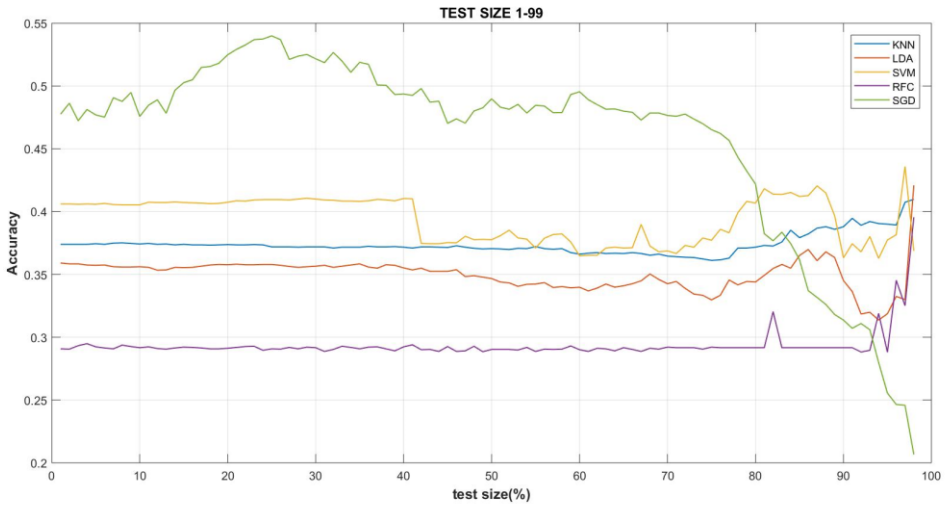


Figure 6.1. Accuracy plot with testing and training size

From the final experiment, it can be seen that with some ripples, the SGD classifier is doing quite well. However, when test size is very high (>90%) and training size is very low (<10%) SGD classifier's accuracy falls drastically. On the other side, when the training size is very small, the kNN classifier with Euclidean distance (when $k = 5$) output has been nearly stable all the time, even more, improved in performance. It is very important that in real cases, typically very few training data are available. In that case, the kNN classifier would be performed consistently.

Additional Experiment (i)

Inspection of transient periods of channels

Status: Trained as Multiple combinations, tested as multiple combinations

Additional experiment (i) was conducted to verify whether or not there are any clear transient phases in the channels. To do this, the data set (Data Crowded) is divided into several parts: transient and stable. It was thought that at the beginning of each channel, there could be some transient periods for all the electrodes in all the rooms. The total dataset is around 600 seconds long. In room 2 with electrode 5 for both empty and crowded situations, transient data consists of about the first 100 seconds of whole data (see Figure 5.3) and stable data contains the rest of the data (i.e. 101s to 600s approximately). The table 6.6 indicates the outcomes of the additional experiment (i).

Table 6.6. Results of additional experiment (i)

Classifiers	Training Size	Classification Accuracy	Status
kNN	75%	80.14%	Trained as stable test as transient
LDA		86.42%	
SVM		78.57%	
RFC		76.42%	
kNN	75%	89.95%	Trained as transient test as stable
LDA		88.46%	
SVM		85.96%	
RFC		81.50%	

In this experiment, all the classifiers provided more than 80% prediction accuracy. As the accuracy is quite high, therefore it is proved that there were no significant transient periods among the channels.

Additional Experiment (ii)

Combined both dataset (Data Empty and Crowded)

Status: Trained as both, tested as Crowded

Another additional experiment carried out during the research. Training and testing data were collected from the same locations at different environmental conditions. The experiment was performed by combining both datasets (Data Empty and Crowded) and training the model by combining the dataset and testing with Data Crowded. Table 6.7 shows the results of an additional experiment (ii).

Table 6.7. Results of additional experiment (ii)

Classifiers	Training Size	Classification Accuracy	Status
kNN	80%	99.97%	Trained as combined dataset /Tested as Crowded
kNN	20%	96.86%	
LDA	80%	92.70%	
LDA	20%	86.17%	
SVM	80%	96.41%	
SVM	20%	90.35%	
RFC	80%	99.90%	
RFC	20%	88.78%	
SGD	80%	99,86%	
SGD	20%	95.12%	

It is clearly seen that, if a model is trained with both environments (Empty and Crowded), it can more accurately predict locations. This result invites for an additional semi-crowded environment dataset. If the third dataset (semi-crowded) was available, how the model reacts with a new dataset could be checked. From the experience of this experiment, the third dataset could be expected to be more accurately predicted locations (> 53 percent) than the outcome of experiment 5.

6.2 Error Sources and Limitations

Despite having very good accuracy over several experiments, there may still be some problems in the process as a whole. Importantly, IMS reading is not stable in time, i.e. changes during measurements and different times of the day. Some of the IMS channels are linked, i.e. channels from different locations show the same values that mislead the classifiers. In the open spaces, airflow is not restricted therefore, measurement results changed in between the two measurements. There are also several error sources in the computational stage. It is very important for the machine learning algorithm to label the training data. If any unlabeled data is provided in the training stage, it will not be smooth over time. In the matching phase, there could be overfitting and underfitting issues. Furthermore, techniques of reducing dimensionality may not fit for all datasets.

In addition, training data and testing data are split randomly; therefore, classification outputs in each execution are not stable. Furthermore, the prediction accuracy varies from the classifier to the classifier.

Some limitations may be highlighted. The variation of IMS reading over time (e.g. day, months, year, and various seasons) could not be checked. This information could be essential for carrying out reliable aroma fingerprint-based localization. Furthermore, information about IMS reading variability might help to determine the condition when training data should be captured [39]. Furthermore, where airflow is restricted (e.g. caves, mines, tunnels), aroma-based localization is presumed to be more viable. Implementing this system in such locations where airborne is continuously circulating and altering is very difficult. It is, therefore, necessary to frequently update the training data.

6.3 Summary of Results

Eight experimental outcomes are all presented in the observation section together. Potential outcomes are obtained from Experiments 1, 2, 3 and 5. kNN classifier with Euclidean distance ($k = 5$) provides better results at the start (i.e. 38 percent). Furthermore, it appears that Data Crowded is more potential than Data Empty. Training with crowded data has accomplished better outcomes in all instances than empty data. Predictive accuracy is strongly linked to the size of training/testing data. The Dimensionality Reduction Technique (PCA) works well with the SVM classifier, where accuracy reached up to 41 percent. Finally, using the more advanced method, at certain conditions, SGD accomplished a predictive accuracy of up to 53 percent. While SGD offers maximum accuracy among all classifiers, but the stable output of kNN with Euclidean distance makes it more convenient to use in any situation.

6.4 Applications

The author is very hopeful about the outcomes of the studies that aroma fingerprinting-based localization which can be achieved in different areas. Localization based on aroma fingerprinting can be the possible method of locating any objects within caves, mines, tunnels, etc. In addition, it can be used to monitor and locate major faults in industrial equipment [60]. (e.g. Gas leakage).

7. CONCLUSIONS

This section includes closing remarks on the ideas and findings of the thesis. Furthermore, relevant future work is also discussed.

7.1 Conclusion

This thesis presented a study on the GNSS denied indoor localization scheme. The primary goal of this thesis was to study the systems that could operate without any external infrastructure being used. Furthermore, the aim was to study various sensors and algorithms necessary for implementing such a system. Different signal properties (e.g. RSSI, ToA, TDoA, AoA) and positioning algorithms (e.g. trilateration, proximity, fingerprinting) are discussed in this thesis. Locating and identifying an object in the indoor environment can be done by using light signals, sound intensity, radio signals, aroma (ion mobility), magnetic fields and other information from sensors collected by mobile phone or any smart devices equipped with modern sensors (Bluetooth, WIFI, infrared, accelerometer, gyroscope, etc.) .In order to introduce an infrastructure-less based positioning algorithm, an alternative method based on aroma fingerprinting has been chosen, which was earlier studied in [39]. Only a few research works were carried out in this selected area. Considering the main thoughts and suggestions from [39], it is possible to improve localization accuracy by using advanced machine learning algorithms. A software-based model was created based on the given datasets from [39]. Python language was used as the implementation platform. A supervised machine learning algorithm was used in the implementation part. Different classifiers have been screened to find one that is best in terms of stability and accuracy. In normal condition, the kNN classifier with Euclidean distance correctly predicted locations about 38 percent of times. Moreover, it is noticed that training and testing size has an important effect on the accuracy of a model. It is observed that from the results of various experiments, Data Crowded has more information about the environment than Data Empty. The further experiment was conducted by training the model with Data Crowded and testing the model with Data Empty. Furthermore, Principal Component Analysis (PCA) is used for removing correlations between variables and transforming them into uncorrelated variables. The first principle component accounts for as much data variation as possible, and as much of the remaining variation as possible is accounted for by each successor element Using PCA, Support Vector Machine (SVM) correctly predicted location around 41 percent of the time.

In the final portion of the thesis, the Stochastic Gradient Descent (SGD) classifier was implemented, and the remarkable outcome was achieved. SGD requires only a certain number of iterations to determine the appropriate response to reduce computing costs and time. The final test was very straightforward, only the model trained with Data Crowded, and then normalized it using "StandardScaler" and lastly used SGD classifier, no PCA and other methods implemented with the SGD experiment. The maximum accuracy attained by stochastic gradient descent is 53 percent, which is the highest score in this research from any classifier.

Infrastructure-less-based positioning techniques will play a vital role in indoor positioning, autonomous vehicular positioning, virtual reality, and emergency rescue mission. Localization using aroma fingerprinting can add a new dimension for positioning in GNSS denied areas. Furthermore, the suggested techniques can support a satisfactory prediction with an affordable set-up for instances of general use. However, some problems still need to be resolved before using aroma fingerprinting as a trustworthy technique of localization. Moreover, no single algorithm wins all the time in classification problems. Different classifiers respond individually with different datasets. The suggested model in the thesis may provide separate outputs with a distinct dataset. In conclusion, the IMS base electronic noses (eNoses) system may have enormous potential in localization applications once the limitations concerning the composition and processing errors are eliminated.

7.2 Future Work

The results shown in the studies are based on two particular datasets and five classifiers. However, this study can be further expanded by conducting additional experiments on data collection and processing. Data collection was conducted under only two environmental circumstances: Crowded and Empty environment. Aroma fingerprints from various environmental conditions such as semi-crowded, different times of the day, months and seasons can be collected. This assessment will assist to get an idea of how the fingerprint has changed over time as well as revealing the best times for collecting training data. Several eNoses IMS measurements can be evaluated and compared from the same locations to investigate the diversity of the device. In addition, to improve the identification efficiency of the measuring tool, it is necessary to replace fewer sensitive electrodes. The AdaBoost algorithm can be tested together with other classifiers. In data science, the AdaBoost algorithm has gained enormous popularity. AdaBoost classifier

combines a set of classifiers that are weak or poorly conducted and then create a powerful one. The accuracy of the new classifier will be higher than the individual classifier. AdaBoost attributes weight to each training data during the training phase. Greater weight is allocated to a misclassified item so that it comes up with greater probability in the next classifier's training subset. The weight is allocated to the classifier on the basis of accuracy after each classifier is trained. Higher weight is allocated to a more precise classifier in order to have more effects on the final results.

REFERENCES

[1] C. Jekeli, Inertial Navigation Systems with Geodetic Applications.Berlin, Germany: Walter de Gruyter, 2001

[2] K. R. Britting, Inertial Navigation Systems Analysis. NewYork: Wiley-Interscience, 2010

[3] A. Solin, S. Cortes, E. Rahtu and J. Kannala, "Inertial Odometry on Handheld Smartphones," 2018 21st International Conference on Information Fusion (FUSION), Cambridge, 2018, pp. 1-5. doi: 10.23919/ICIF.2018.8455482.

[4] W. Sakpere., Adeyeye-Oshin, M. and Mlitwa, N.B.W. (2017). A state-of-the-art survey of indoor positioning and navigation systems and technologies. South African Computer Journal 29(3), 145–197. https://doi.org/10.18489/sacj.v29i3.452.

[5] M. Kaluža, K. Beg, B. Vukelić: Analysis of an indoor positioning systems Zbornik Veleučilišta u Rijeci, Vol. 5 (2017), No. 1, pp. 13-32

[6] F. Zafari, A. Gkelias and K. K. Leung, "A Survey of Indoor Localization Systems and Technologies," in *IEEE Communications Surveys & Tutorials*. doi: 10.1109/COMST.2019.2911558

[7] D. Bucur., & M. B. Kjærgaard (2008). GammaSense: Infrastructureless Positioning Using Background Radioactivity. In D. Roggen, C. Lombriser, G. Tröster, G. Kortuem, & P. Havinga (Eds.), Smart Sensing and Context: Third European Conference, EuroSSC 2008, Zurich, Switzerland, October 29-31, 2008, Proceedings (pp. 69-82). Berlin, Heidelberg: Springer.

[8] L.T. Nguyen., Y. Zhang. (2012) Probabilistic Infrastructureless Positioning in the Pocket. In: Zhang J.Y., Wilkiewicz J., Nahapetian A. (eds) Mobile Computing, Applications, and Services. MobiCASE 2011, Los Angeles, CA, USA, October 24-27, 2011.

[9] P. Davidson & R. Piche (2017). A Survey of Selected Indoor Positioning Methods for Smartphones. IEEE Communications Surveys and Tutorials, 19(2), 1347-1370. DOI: 10.1109/COMST.2016.2637663

[10] N. Lee & S. Ahn & D. Han. (2018). AMID: Accurate Magnetic Indoor Localization Using Deep Learning. Sensors. 18. 1598. 10.3390/s18051598.

[11] K. Wroble, "Performance Analysis of Magnetic Indoor Local Positioning System" (2015). *Master's Theses*. 609. P-5

[12] Magnetic flux density [www.goudsmitmagnets.com/en/wiki/66/flux-density-magnetic-flux-density-b], [Online accessed: 18/4/2019]

[13] J.A. Shockley., J.F. Raquet. Navigation of Ground Vehicles Using Magnetic Field Variations. Navigation. 2014; 61:237–252. doi: 10.1002/navi.70.

[14] HS. Kim, W. Seo, KR. Baek. Indoor Positioning System Using Magnetic Field Map Navigation and an Encoder System. *Sensors (Basel)*. 2017;17(3):651. Published 2017 Mar 22. doi:10.3390/s17030651

[15] Inertial Navigation System [www.sciencedirect.com/topics/engineering/inertial-navigation-system] [Online accessed: 23/4/2019]

[16] Inertial Navigation System (INS) [www.skybrary.aero/index.php/Inertial_Navigation_System_ (INS)] [Online accessed: 23/4/2019]

[17] O. J. Woodman, Technical reports published by the University of Cambridge 2007, ISSN 1476-2986, http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-696.pdf

[18] Inertial Navigation Systems Specifications [www.globalspec.com/learnmore/sensors_transducers_detectors/tilt_sensing/inertial_gyros], [Online accessed: 24/4/2019]

[19] S. Aditya & A. F. Molisch & H. Behairy (2018). A Survey on the Impact of Multipath on Wideband Time-of-Arrival-Based Localization. Proceedings of the IEEE. PP. 1-21. 10.1109/JPROC.2018.2819638.

[20] S. Holm, Ultrasound positioning based on time -of-flight and signal strength. Presentation to be given by Sverre Holm at 3rd International Conference on Indoor Positioning and Indoor Navigation (IPIN), 13-15th November 2012 at the University of New South Wales, Sydney, Australia.

[21] A. Ward, A. Jones, and A. Hopper, “A new location technique for the active office,” IEEE Personal Communications, vol. 4, no. 5, pp. 42–47, Oct 1997.

[22] N. Priyantha, A. Chakraborty, and H. Balakrishnan, “The cricke location -support system,” in Proc. 6th Ann. Int. Conf. Mobile Computing and Networking. ACM, 2000, pp. 32–43.

[23] M. Ruffo., M. D. Castro, L. Molinari, R. Losito, A. Masi, J.W. Kovermann, & L. Rodrigues (2014). New Infrared Time Of-Flight Measurement Sensor for Robotic Platforms. 20th IMEKO TC4 International Symposium and 18th International Workshop on ADC Modelling and Testing, Benevento, Italy, September 15-17, 2014

[24] A. Badawy, T. Khattab, D. Trinchero, T. ElFouly and A. Mohamed, "A Simple Angle of Arrival Estimation System," *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, San Francisco, CA,2017,pp.1-6.doi: 10.1109/WCNC.2017.7925867

[25] Finding Location with Time of Arrival and Time Difference of Arrival Techniques [https://sites.tufts.edu/eeseniordesignhandbook/files/2017/05/Fire-Brick_OKeefe_F1.pdf] [Online: Accessed 17-05-2019]

[26] F. Zafari, A. Gkelias and K. K. Leung, "A Survey of Indoor Localization Systems and Technologies," in *IEEE Communications Surveys & Tutorials*. doi: 10.1109/COMST.2019.2911558

[27] Z. Yang, Z. Zhou, and Y. Liu, “From RSSI to CSI: Indoor localization via channel response,” ACM Computing Surveys (CSUR), vol. 46, no. 2, p. 25, 2013.

[28] P. Kumar, L. Reddy, and S. Varma, “Distance measurement and error estimation scheme for RSSI based localization in Wireless Sensor Networks,” in Wireless Communication and Sensor Networks (WCSN),2009 Fifth IEEE Conference on, pp. 1–4, IEEE, 2009.

[29] Prof. M Torlak, EE4367 Telecom. Switching & Transmission, lecture slides [https://docplayer.net/10036529-Ee4367-telecom-switching-transmission-prof-murat-torlak.html] [Online: Accessed 20-08-2019]

[30] H. Liu., H. Darabi, P. Banerjee, & J. Liu (2007). Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, *37*(6), 1067–1080. https://doi.org/10.1109/TSMCC.2007.905750

[31] Y. Gu, A. Lo, & I. Niemegeers (2009). A survey of indoor positioning systems for wireless personal networks. *IEEE Communications Surveys & Tutorials*, *11*(1), 13–32. https://doi.org/10.1109/ SURV.2009.090103

[32] W. Sakpere, M. Adeyeye-Oshin and N.B.W. Mlitwa,. (2017). A state-of-the-art survey of indoor positioning and navigation systems and technologies. *South African Computer Journal 29*(3), 145–197. https://doi.org/10.18489/sacj.v29i3.452

[33] H. Liu., H. Darabi, P. Banerjee, & J. Liu (2007). Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, *37*(6), 1067–1080. https://doi.org/10.1109/TSMCC.2007.905750

[34] T.T J. Brooks, H. HC Bakker, and K. A. Mercer2 W. H. Page. "A Review of Position Tracking Methods. 1st International Conference on Sensing Technology November 21-23, 2005 Palmerston North, New Zealand.

[35] R. Mautz (2012). *Indoor positioning technologies* (Habilitation thesis, ETH Zurich).

[36] A. Mukhopadhyay, & A. Mallisscry (2018). TELIL: A Trilateration and Edge Learning based Indoor Localization Technique for Emergency Scenarios. 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 6-13.

[37] J. Xiao, Z. Liu, Y. Yang, D. Liu & X. Han (2011). Comparison and analysis of indoor wireless positioning techniques. In *International Conference on Computer Science and Service System* (pp. 293–296). Nanjing: IEEE. https://doi.org/10.1109/CSSS.2011.5972088

[38] L. Brás, N.B. Carvalho, P. Pinho, L. Kulas & K. Nyka (2012). A review of antennas for indoor positioning systems. *International Journal of Antennas and Propagation*, *2012*(Article ID 953269), 1–14. https://doi.org/10.1155/2012/953269

[39] P. Müller, J. Lekkala, S. Ali-Löytty & R. Piche (2017). Indoor Localisation using Aroma Fingerprints: A First Sniff. In 2017 14th Workshop on Positioning, Navigation and Communications (WPNC) Bremen, Germany: IEEE. DOI: 10.1109/WPNC.2017.8250046

[40] J. Yim, C. Park, J. Joo, & S. Jeong (2008). Extended Kalman filter for wireless LAN based indoor positioning. *Decision Support Systems*, *45*(4), 960–971, Elsevier Science Publishers B. V. Amsterdam, The Netherlands, The Netherlands, Publication Date 2008-11-01. https://doi.org/10.1016/j.dss.2008.03.004

[41] O. Gursoy & P. Somervuo & T. Alatossava (2009). Preliminary study of ion mobility based electronic nose MGD-1 for discrimination of hard cheeses. Journal of Food Engineering. 92. 202-207. 10.1016/j.jfoodeng.2008.11.002.

[42] ChemPro100i Handheld Chemical Detector [www.environics.fi/product/chempro100i/] [Online: Accessed 17-05-2019]

[43] Owlstone Nanotech White Paper info. [owlstonenanotech.com/rs/owlstone/images/FAIMS%20Whitepaper.pdf] [Online: Accessed 17-05-2019]

[44] ChemPro100i Handheld Chemical Detector [brj.dk/wp-content/uploads/2013/09/CP100-data-sheet-english.pdf] [Online: Accessed 17-05-2019]

[45] Machine Learning [se.mathworks.com/discovery/machine-learning.html] [Online: Accessed 17-05-2019]

[46] A Tour of The Most Popular Machine Learning Algorithms [machinelearningmastery.com/a-tour-of-machine-learning-algorithms/] [Online: Accessed 18-05-2019]

[47] Reinforcement learning [www.geeksforgeeks.org/what-is-reinforcement-learning] [Online: Accessed 19-05-2019]

[48] Heikki Huttunen, Pattern Recognition and Machine Learning, lecture slides [heikki.huttunen@tuni.fi] L-4

[49] Introduction to machine learning [aldro61.github.io/microbiome-summer-school-2017/sections/basics] [Online: Accessed 20-05-2019]

[50] K Nearest Neighbours- Introduction to Machine Learning Algorithms [towardsdatascience.com/k-nearest-neighbours-introduction-to-machine-learning-algorithms-18e7ce3d802a] [Online: Accessed 23-05-2019]

[51] Support Vector Machine-Introduction to Machine Learning Algorithms [towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47] [Online: Accessed 23-05-2019]

[52] Understanding Random Forests Classifiers in Python [www.datacamp.com/community/tutorials/random-forests-classifier-python] [Online: Accessed 24-05-2019]

[53] Gradient descent and stochastic gradient descent from scratch [gluon.mxnet.io/chapter06_optimization/gd-sgd-scratch.html] [Online: Accessed 25-05-2019]

[54] A step by step explanation of Principal Component Analysis [towardsdatascience.com/a-step-by-step-explanation-of-principal-component-analysis-b836fb9c97e2] [Online: Accessed 28-05-2019]

[55] A. Solin, S. Cortes, E. Rahtu, and J. Kannala, “PIVO: Probabilistic inertial-visual odometry for occlusion-robust navigation,” in Proceedings of WACV, 2018

[56] A. Solin, S. Särkkä, J. Kannala and E. Rahtu, "Terrain navigation in the magnetic landscape: Particle filtering for indoor positioning," 2016 European Navigation Conference (ENC), Helsinki, 2016, pp. 1-9.doi: 10.1109/EURONAV.2016.7530559

[57] O. Oguejiofor, V. Okorogu, A. Adewale, B. Osuesu, “Outdoor Localization System Using RSSI Measurement of Wireless Sensor Network”, International Journal of Innovative Technology and Exploring Engineering, vol. 2, Jan.2013

[58] Regression Definition [www.investopedia.com/terms/r/regression.asp] [Online: Accessed 25-06-2019]

[59] K Nearest Neighbors - Classification [www.saedsayad.com/k_nearest_neighbors.htm] [Online: Accessed 27-06-2019]

[60] V. H. Bennetts, A. J. Lilienthal, P. P. Neumann, and M. Trincavelli, Mobile robots for localizing gas emission sources on landfill sites: is bio-inspiration the way to go? Frontiers in Neuroengineering, vol. 4, pp. 1-12, January 2012.

[61] Global Navigation Satellite Systems (GNSS) [https://www.gps.gov/systems/gnss] [Online: Accessed 16-04-2019]

[62] European Global Navigation Satellite Systems [https://www.gsa.europa.eu] [Online: Accessed 16-04-2019]

[63] The reference for Global Navigation Satellite Systems [https://gssc.esa.int/navipedia/index.php/Main_Page] [Online: Accessed 16-04-2019]

[64] An Introduction to GNSS [https://www.novatel.com/an-introduction-to-gnss] [Online: Accessed 16-04-2019]