NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCE
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATION

MSc THESIS

Artificial neural network techniques for estimation of indoor LiFi user position

Dilnoza T. Ibragimova

Supervisor (or supervisors): **Panagiotis Stamatopoulos**, Assistant Professor

ATHENS

MAY 2022

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Εκτίμηση της θέσης χρήστη LiFi σε εσωτερικούς χώρους με χρήση τεχνητών νευρωνικών δικτύων

Dilnoza T. Ibragimova

Επιβλέποντες: **Παναγιώτης Σταματόπουλος**, Επίκουρος Καθηγητής

ΑΘΗΝΑ

May 2022

MSc THESIS

Artificial neural network techniques for estimation of indoor LiFi user position

Dilnoza T. Ibragimova
S.N.: 1200003

SUPERVISOR: **Panagiotis Stamatopoulos**, Assistant Professor

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Εκτίμηση της θέσης χρήστη LiFi σε εσωτερικούς χώρους με χρήση τεχνητών νευρωνικών δικτύων

Dilnoza T. Ibragimova

S.N.: 1200003

ΕΠΙΒΛΕΠΟΝΤΕΣ: **Παναγιώτης Σταματόπουλος,** Επίκουρος Καθηγητής

ABSTRACT

This Thesis studies a novel LiFi optical communication system, the 3D position and orientation of users' devices depending on collected, labelled, and tested dataset by using kernel-based Artificial Neural Network approaches. Both the collection and labelling of a dataset and the used models are explained. The ANN algorithms such as CNN, MLP, and SVM are presented and the results are compared with KNN output in terms of performance, such as utilized time, bit error rate, accuracy, and average estimate error.

The whole process was divided into several steps like data collection, training model, and testing. The data collection process utilizes MATLAB code based on rotation angle estimation, truncated Laplace distribution considering both LOS and NLOS, while the training model includes choosing the best mapping model to reach optimal output in terms of positioning, orientation angles of user equipment, and received SNR vector. In the testing phase, it is obvious that models are fed by unseen and unexplored data and then the correctness of results is estimated.

Subject area: Device positioning

Keywords: KNN, CNN, SVM, ANN, Machine Learning

ΠΕΡΙΛΗΨΗ

Αυτή η εργασία μελετά ένα νέο σύστημα οπτικών επικοινωνιών LiFi για την εκτίμηση της τρισδιάστατης θέσης και του προσανατολισμού των συσκευών των χρηστών, με βάση συλλεγμένα, επισημασμένα και ελεγμένα δεδομένα, χρησιμοποιώντας προσεγγίσεις τεχνητών νευρωνικών δικτύων που βασίζονται σε πυρήνα. Περιγράφονται τόσο η συλλογή και η επισήμανση ενός συνόλου δεδομένων, όσο και τα χρησιμοποιούμενα μοντέλα. Παρουσιάζονται οι αλγόριθμοι ANN, όπως CNN, MLP και SVM και τα αποτελέσματα συγκρίνονται με αυτά του KNN ως προς την απόδοση, όπως ο χρόνος που χρησιμοποιείται, ο ρυθμός σφάλματος bit, η ακρίβεια και το μέσο σφάλμα εκτίμησης. Η όλη διαδικασία χωρίστηκε σε διάφορα στάδια, όπως συλλογή δεδομένων, μοντέλο εκπαίδευσης και έλεγχο. Στη διαδικασία συλλογής δεδομένων χρησιμοποιείται κώδικας MATLAB που βασίζεται σε εκτίμηση γωνίας περιστροφής, περικομμένη κατανομή Laplace με βάση τόσο το LOS όσο και το NLOS, ενώ το μοντέλο εκπαίδευσης περιλαμβάνει την επιλογή του καλύτερου μοντέλου απεικόνισης για την επίτευξη βέλτιστης απόδοσης όσον αφορά τη θέση, τις γωνίες προσανατολισμού του εξοπλισμού χρήστη και το λαμβανόμενο SNR διάνυσμα. Στη φάση του ελεγχου, είναι προφανές ότι τα μοντέλα τροφοδοτούνται από άγνωστα και ανεξερεύνητα δεδομένα και στη συνέχεια εκτιμάται η ορθότητα των αποτελεσμάτων.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: π.χ. Device positioning

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: π.χ. KNN, CNN, SVM, ANN, Machine Learning

To my family.

ΕΥΧΑΡΙΣΤΙΕΣ (ή AKNOWLEDGMENTS)

I thank my thesis supervisor, Prof. Panagiotis Stamatopoulos, for guiding my work, I am grateful for his support, helpful advice, constructive criticism, patience, and encouragement. I am thankful to my co-supervisor Prof. D.Syvridis for his suggestions and help on the theme.

Moreover, I am grateful to the research students of the Smartnet and Pixnet programs. I would like to thank Afnan Afnan Amayreh, Mariano Devigili, and Mahmoud Esawi for their support both in working on the thesis and as a friend in hard times. I want to thank my parents who always were encouraging and supporting me, who make me who I am.

Additionally, I want to say grateful words to Tetyana Gordienko and Zorina Bousboura for helping with all administrative issues and being helpful and friendly.

Finally, I want to express my gratitude to the Smartnet program and especially Prof. Sygletos Stylianos for this great opportunity to be enrolled and be a student of the Erasmus Mundus Joint Degree program

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Over the last decade, network complexity has been under a tremendous growth with a rapid increase of Internet and telecommunication. According to [1], the percentage of the compound annual growth rate of wireless traffic was defined by 60% and the number will double every year for the next twenty years [2]. The integration of industrial automatization for different fields like smart cars and homes, e-health, virtual reality, and most importantly the IoT and 5G leads to huge bandwidth demand. Both current and beyond wireless networks need efficient solutions with high security, low-cost equipment, reasonable energy consumption, and flexibility. In addition, wide-coverage, high data rates, and low latency communication are extremely demanded to provide a significant quality for IoT services, which require immediate response in any location at any time.

The capability of telecommunications operators to detect the location and data of mobile terminals at communications stations -- access points (APs) and base stations (BSs) has become a crucial component in establishing next-generation communication systems. Such knowledge helps for a more accurate assessment of wireless link quality, which can improve with resource management and the development of new location-based applications.

Positioning systems are widely used in last years in wide range of fields likewise portable devices, aircraft, navigation applications in vehicles, etc. Traditional positioning techniques like GPS accept inaccuracy in outdoor environment up to meter. Unlike to this, indoor environment requires high level of accuracy counting millimeters. Moreover, indoor environments suffer from inaccuracy in data as a result of a vast range of obstacles since received signals have to overcome clouds, ceilings, walls, etc. [3].

The fact that objects to be located frequently travel through confined areas like corridors, staircases, aisles, and specialized rooms, or stay in a position with many surrounding barriers that remain out of sight, and this distinctive aspect make positioning in indoor environments challenging. Wireless networks like WiFi and Bluetooth have already provided a solution that is widely integrated. However, considering the access points limitation and other imperfections these networks are not enough.

LiFi was mentioned for the first time by Dehghani Soltani et al. in 2011 [4]. While LiFi is a novel wireless system and an extension of VLC with the prospective of extremely high speed at low cost, Machine Learning is an essential key booster almost for all modern

systems and spheres. By synthesizing these two components we can receive reliable positioning data. Unfortunately, most of the works based on this idea use ideal conditions with knowing all access points and user devices while this thesis utilizes knowledge of at least one access point. RSS estimation metric was chosen using experience of previous publications [5] and by calculating the angle at the receiver so location can be obtained.

Machine Learning is a field of Artificial Intelligence which utilized in most systems including positioning applications. In [6] several ANN models were tested for indoor positioning such as CNN and MLP and results were compared with KNN, and superiority of ANN models was proved. However, there is still algorithms to be investigated and procedure time to be improved.

Consequently, for accurate positioning and orientation assessment, a new reliable solution is offered. SVM model with ANN modification was developed and tested on collected dataset and comparison between all models was pictured.

2 BACKGROUND

This chapter is composed by two main sections: Section 2.1 contains a description of the LiFi technology whereas in Section 2.1 a state-of-the-art of the machine learning techniques is presented.

2.1 LiFi Protocol

LiFi stands for Light Fidelity and was first specified by H.Haas in 2011 and promises to be a disruptive technology that can force traditional IoT to better performance, boost wireless technologies, and significantly improve the lighting industry.

LiFi conceived as a novel wireless bi-directional communication technology and as an extension of VLC (visible light communication) to achieve high-speed, secure communication and illumination. LiFi utilizes infrared and visible light spectrum.

2.1.1 LiFi concept

LiFi is a bi-directional, secure, high-speed wireless networking solution that facilitates user mobility. Multiple access points (APs) are typically required to construct dense optical attocellular networks. Co-channel interference can be regulated in more accurate manner than in RF communication systems, thus cell sizes can be ultra-small with radii in the meter range.

LiFi also gains from a vast volume of free spectrum and allows to drive the small-cell concept to new heights that are difficult to achieve in radio frequency (RF). In fact, LiFi system reach three orders of magnitude improvements in terms of area spectrum efficiency, reaching tens of Gbps/m2. Multiple access points build a wireless network of very small optical attocells with flawless handover. As a result, LiFi leads to entire user mobility, forming a new layer within current heterogeneous wireless networks [7].

LiFi exploits light emitting diodes (LED) for high speed wireless communication. LiFi signals can be contained locally thanks to LEDs' inherent beamformers, and because the signals are obstructed by opaque walls, CCI can be successfully handled, and physical layer security can be strengthened. LiFi utilizes IR for uplink and visible light for downlink, this approach allows avert any effect on the intensity of luminosity and minimize interference in the indoor environment. It is also worth emphasizing that the key difference is that VLC is a wireless single point-to-point communication system while LiFi provides multi-user access and user mobility.

The data rate scales with the obtained signal-to-noise ratio by utilizing an orthogonal frequency division multiplexing (OFDM)-type intensity modulation (IM)/direct detection (DD) modulation method [8].

LiFi is an important boost factor for the fusion of two significant sectors like the lightning industry and wireless communication industry. Consequently, one of the LiFi advantages is that Access Point can be combined with LED illumination. However, it is an optional function and while the infrared spectrum is perfect for uplink communication with LiFi utilization, lightning can be fully separated with no damage quality [9].

The fact that LiFi uses optical resources turns it into a very attractive technology since it means free-license and plentiful. Also, LiFi leads to shifting the paradigm by changing cm-wave communication to nm-wave communication. For instance, in [10] the experiment was created where received data at a speed of 1.1 Gbps over the distance of 10 m with an LED of only 4.5 mW optical output power.

The following are some of the benefits of LiFi:

- three orders of magnitude increased data densities [11];
- unique properties to improve physical layer security;
- use in intrinsically safe environments such as petrochemical plants and oil platforms where RF is often prohibited; and
- with the advent of power-over-ethernet (PoE) and its use in lighting, there exists the opportunity [12].

2.1.2 LED advantages

In recent years, positioning systems based on light emitting diodes (LEDs) have appeared, which use visible light instead of radio frequency (RF). Previously, LED was renowned for its promising communication use, specifically visible light communication (VLC) [13]. LED's numerous vital properties are the key to many applications.

These characteristics are:

- Energy efficiency. General and domestic lightning consumption can be reduced by 80%.
- Generate less heat
- Longer service lifetime

- Free from hazardous substances
- Green technology that helps to reduce Co2 emission and electricity consumption [14,15,16]

2.2 Machine Learning techniques

Machine Learning (ML) is a subcategory of artificial intelligence that describes the process through which computers gain pattern recognition, or the ability to continuously study from and predict outcomes based on data, and then make improvements without being explicitly programmed to do so [17]. ML can be roughly categorized as:

- Supervised
- Unsupervised

2.2.1 Supervised Learning

In supervised learning (SL) - the main characteristic is that the model is trained on a known dataset with labels and as the next step the model is fed with a new unknown feature to recognize a solution. In these kinds of algorithms, there is a given dataset X of instances and labels Y. Algorithms aim to build a function that relates dataset x to labels y [18].

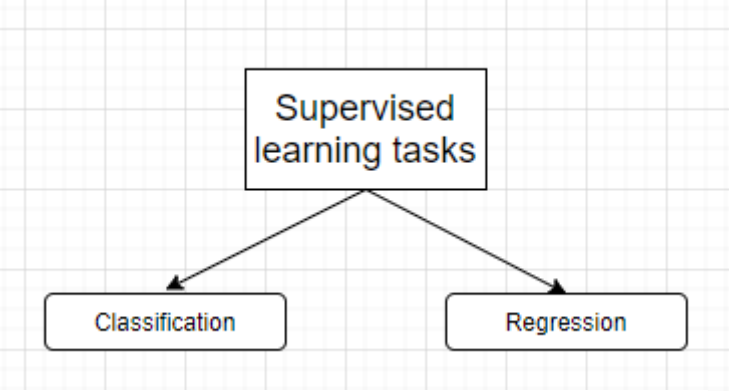

Figure 1 Types of SL tasks

As can be seen from the illustration there are two kinds of tasks in SL: classification and regression. *Classification* is a task where a labeled class is a recognized solution for a given dataset [19]. Meanwhile, *regression is* known as a challenge of predicting constant quantity output for an existing example [20]. Generalization is the main problem faced by Supervised Learning where the model should give the correct output on new data, not only on the given one [21].

Critical aspects of SL are Overfitting and Underfitting. During the training phase, the type and amount of input data have a significant impact on the machine learning model's performance. Overfitting or underfitting occurs when the training data is insufficient. The phenomenon of overfitting occurs when the validation error grows while the training error decreases. This occurs because, rather than learning the real data distribution, the model learns the expected output for each input data. Underfitting issues, on the other hand, occur when a model is unable to learn enough due to a lack of training data [22]. Figure 2 illustrates each situation clearly.

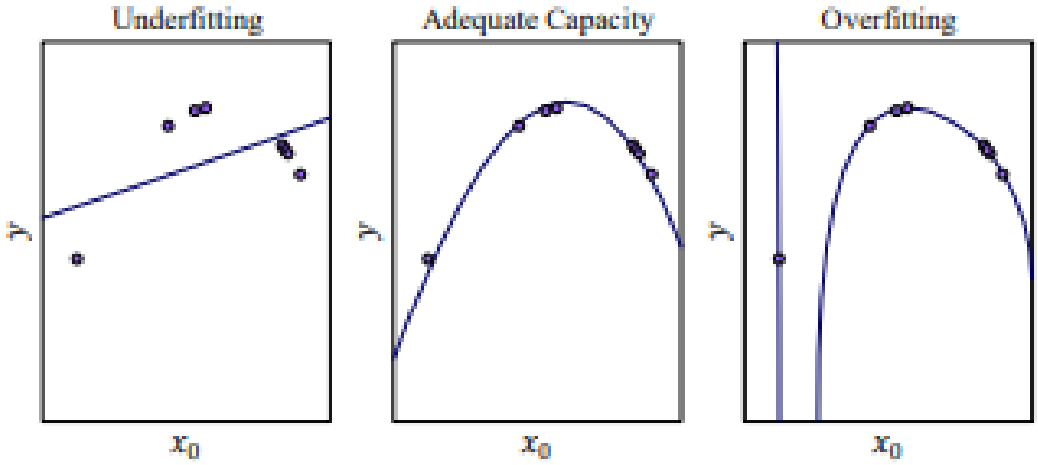

Figure 2 Examples of Underfitting, Adequate Capacity and Overfitting

With the development of SL, large-scale data - already pre-trained - can be found as an open source. For example, CNN datasets or GoogleNet [23].

2.2.2 Unsupervised Learning

Oppositely to SL algorithms, Unsupervised Learning (UL) ones do not use labeled dataset. On the contrary UL techniques detect the connection among the elements of a data set. Algorithms offer solutions without inquiry. The model is trained on an unlabeled dataset to detect subgroups with similar attributes between variables without supervision. This kind of methodology is advantageous when it is necessary to discover patterns in a dataset or it is not feasible to obtain the ground truth labels [24].

Clustering is an unsupervised ML task, which includes the interpretation of input datasets and detecting clusters. The number of clusters can be indicated in some cases [25]. The main clustering types can be listed as follows**:**

- Hierarchical clustering
- K-means clustering
- K-NN (k nearest neighbors)

- Principal Component Analysis
- Singular Value Decomposition
- Independent Component Analysis

Unsupervised Learning algorithms have a great potential for LiFi and wireless network communication. The term anomaly is used when for instance at a certain time and location an unusual traffic is detected. Hierarchical clustering is one of the most efficient ways to help to find anomalies [26].

The table is describing the main differences between the 2 levels of supervision and concludes the information above.

Table 1 Comparison between SL and UL

Basis for comparison	**Supervised**	**Unsupervised**
Training data	Labeled data	Unlabeled data
Preference	Input output mapping (Routine problem)	Clustering, detecting data correlation and new patterns
Area	ML	ML
Optimal strategy	Depend on learning model and the data	Depend on the data and its classification

2.2.3 Machine Learning Models

With the dramatic expansion of AI utilization, indoor positioning systems witness a demand. Defining the location of UE with high speed in real-time requests accuracy which is the basis for indoor environment, service robots, and drones. In [27, 28] KNN, ANN, multiple classifiers, and clustering algorithm performance are discussed in terms of efficiency, accuracy, and robustness. In [29, 30] to overcome the disadvantages and limitations of the KNN algorithm like slow execution time, high error formation was offered a maximum received signal strength recognition (MRR) technique and weighted optimum KNN (WOKNN) algorithm, which is a combination of optimum KNN (OKNN) and weighted KNN (WKNN). In [31, 32] the results of the ELM technique are compared

with KNN and SVM and demonstrate higher accuracy and reasonable speed. Moreover, Kernel is offered as a way of reducing the size of the fingerprint database and thereby exponentially reducing training time. However, the parameters are not suitable for real life. Therefore as an example for the thesis, the paper of H.Haas [33] was chosen. This research builds a dataset that fits real life and can be used as a base.

KNN model: K-nearest neighbor is a supervised learning algorithm applied for classification problem or as a benchmark in classification problem. This approach relies on the nearest neighbors to orientate a receiver. The process includes two phases:

- Offline: The RSS values calculated at the receiver for each LED are used to determine the fingerprints of each position positions within the network's area.
- Online: The Euclidean distances dE between offline RSS values and online RSS values calculated at the receiver during its movement are estimated in the online stage as shown below, where RSS_{Ti} – RSS values obtained in the online phase, i and l are LED index and numbers of LED respectively, and RSS_i – values stored in the previous stage. $\sqrt{\sum_{i=1}^{n}(x_i - y_i)}$

$$d_E = \sqrt{\sum_{i=1}^{l}(RSS_{Ti} - RSS_i)^2}$$

K-nodes that hold the smallest distance are set as a K nearest neighbor of the receiver. The location is calculated by averaging the k-NN coordinates as shown below:

$$x = \frac{\sum_{i=1}^{k} x_i}{k}; \quad y = \frac{\sum_{i=1}^{k} y_i}{k}$$

ANN model: Artificial Neural Network is one of the most popular ML algorithms nowadays. The technique is based on collection connected artificial neurons where each connection is able to transmit the signal to other neurons. The graph below demonstrates ANN concept as a multilayer and fully connected neural network with three input nodes, two hidden layers and one output node.

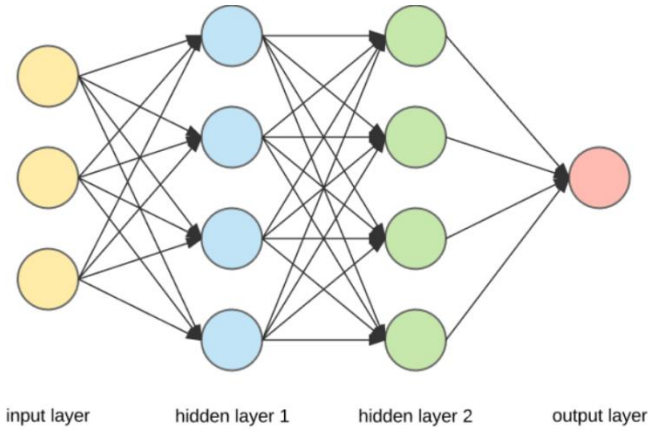

Figure 4: ANN architecture

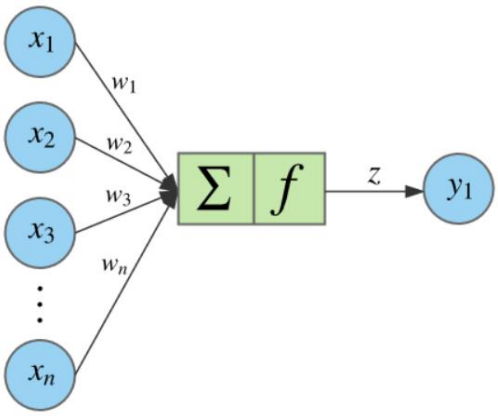

Figure 3: Structure of a node

A given node computes the weighted sum of its inputs, feeds it to an activation function and the output becomes the input for the next layer. By repeating this procedure for each node, the final output is computed. Learning the weights associated with all the edges is what training this deep neural network entails. To apply this scheme for LiFi positioning system, Figure 5 demonstrates the calculation coordinates as an output layer

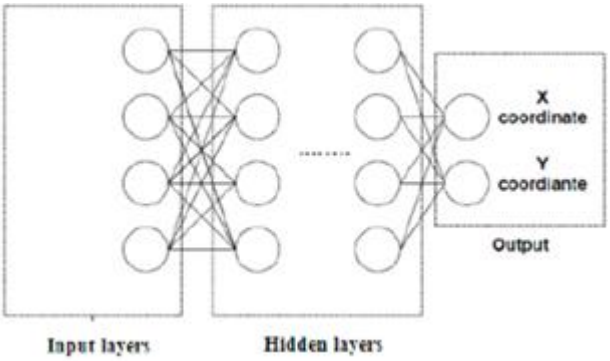

Figure 5: Example of ANN for LiFi positioning system

A state of art review of the field of study, including current developments, controversies and breakthroughs, previous research and relevant background theory. At Masters level it must be analytical and summative, covering methodological issues, research techniques and topics. There may possibly two literature-based chapters, one on methodological issues, which demonstrates knowledge of the advantages and

disadvantages, and another on theoretical issues relevant to the topic/problem. A short, logical summing up of the theme(s) developed in the main text.

MLP is a feedforward ANN which means that information flows through all layers without any backward loops and in one direction, passing through single input, at least one hidden, and finally single output layers. The next expression describes the output of each neuron:

$$y = \varphi\left(\sum_{k=0}^{n} w_k\, x_k\right)$$

where $\varphi(x)$ is an activation function, n – number of neuron inputs, w_k – weight of x_k – input value at k-th neuron. Each activation function is given the total of the weighted inputs with a bias term. The activation function is a mathematical gate that connects a neuron's inputs and outputs. It can be a step function (the output is active if the input value is larger than a threshold), a linear function (the output is the input times some constant factor), or a non-linear function.

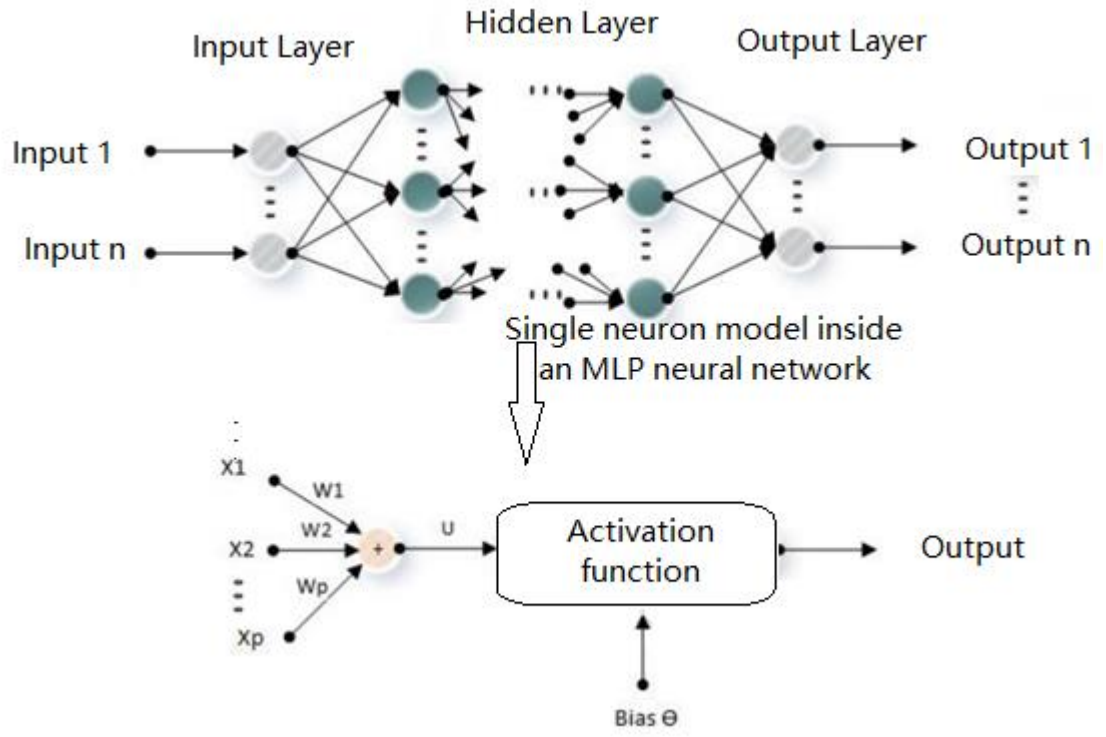

Figure 6: MLP architecture

The non-linear functions enable the model to map the complex interactions between inputs and outputs, which is critical for learning and modelling complex real-world data.

Convolutional Neural Networks (CNN): Data features are extracted via local connections and layer-by-layer computation in the CNN model. This model includes convolutional layer (Conv2D), Max pooling and fully connected layer. The construction of the CNN model is depicted in Figure, where the convolutional layer runs the convolution process with input data and each convolution kernel extracts input data features. The weight sharing method in the convolution operation helps to reduce the number of parameters and the neural network's training difficulty. The weight sharing mechanism boosts the network's training speed, while the pooling layer in the CNN model minimizes the dimension and size of the input data.

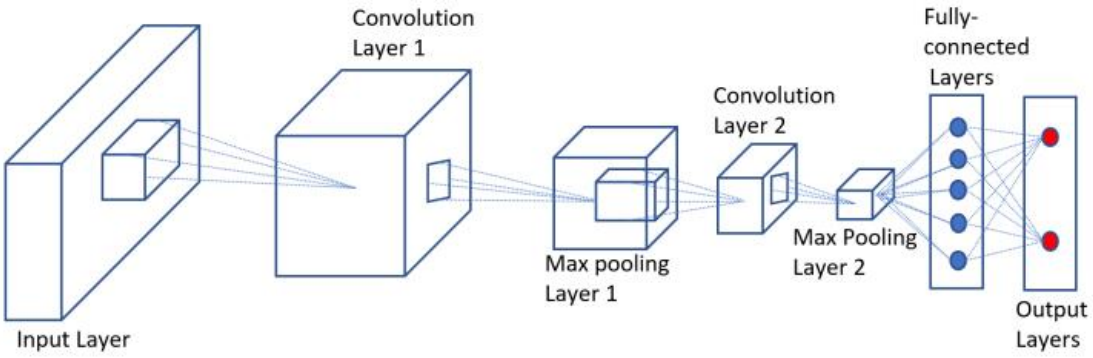

Figure 7: CNN architecture [34]

Support Vector Machines (SVM): SVM is a type of supervised learning model that is widely used to address linear classification problems by maximizing the feature space margin. Because the constructed sample set is linearly inseparable, a kernel function is used to convert a nonlinear problem into a linear problem capable of translating the SVM training samples from the original space to the high-dimensional space where the samples are linearly separable.

This classification is exclusively based on the perception of a collection of the form (x_i, y_i), i:1, n, where xi represents the i-th input and y_i represents the matching output.

The optimization problem for the linear case is expressed as:

$$\begin{cases} Min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \epsilon_i \\ \forall i,\ \mathrm{y}_i(w.\ x_i + b) \geq 1 - \epsilon_i \end{cases}$$

The optimization problem for non-linear classification is expressed as:

$$\begin{cases} Max \sum_{i=1}^{n} \alpha_x - 1/2 \sum_{i,j} \alpha_i \alpha_j y_i y_j\ K\ (x_i, x_j) \\ \forall i,\ 0 \leq \alpha_i \leq C \\ \sum_{i=1}^{n} \alpha_i\ y_i = 0 \end{cases}$$

Where K represents a kernel function that calculates the distance or similarity between the input vector x_i and the stored vector x_j. The Gaussian kernel and the Polynomial kernel are two examples of kernel functions. The SVM's kernel is in charge of converting the input data into the appropriate format. SVM kernels include linear, polynomial, and radial basis functions (RBF). We utilize RBF and the Polynomial function to create a non-linear hyperplane. To distinguish nonlinear classes in complicated applications, more powerful kernels should be used. This transformation can provide accurate classifiers [35].

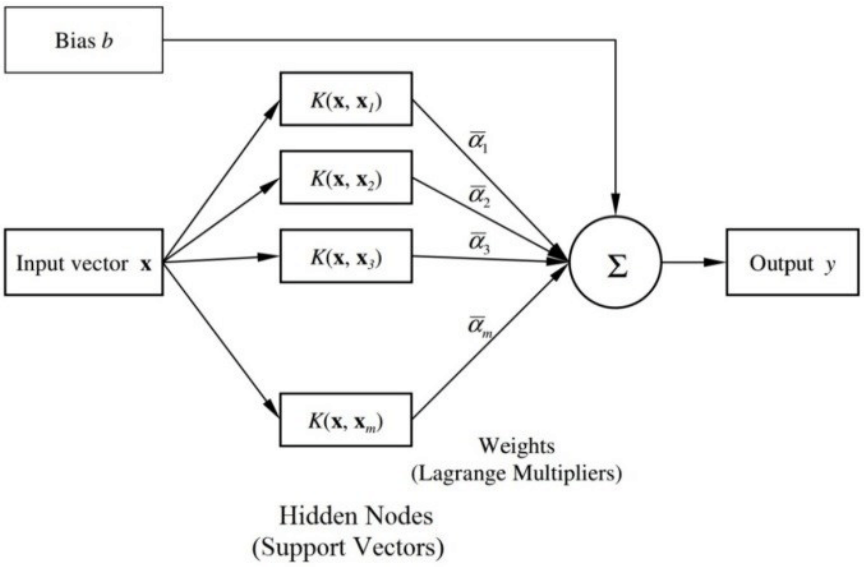

Figure 8: SVM architecture

2.3 Tools

The Golden Rule is that there should be no surprises here and this section should be kept fairly short. Present the conclusions one by one in a logical order. each should be brief and self-contained. Each conclusion must be drawn in a logical order from what has gone before: *fact, fact, fact* therefore *conclusion* based on your analysis and discussion.

2.3.1 ReLu activation functions

ReLU activation functions frequently utilized in ANN networks. The ReLU function reduces computation demand, but at the expense of accuracy. The ReLU function is used because it provides great accuracy at a minimal computational cost. The conventional ReLU function is a piecewise function:

y = 0, if x<0

$y = x$, if $x>=0$

It retains positive values and returns zero for negative inputs. This behaviour can be troublesome depending on the range of inputs. If the majority of the numbers in a batch are positive, too much redundant information is maintained, resulting in greater bit-width computations or overflow for subsequent calculations. If, on the other hand, the majority of the numbers are negative, zero is output for them, and a lot of essential information is lost. This would prevent the network from learning and could even result in the network's death [36].

2.3.2 Batch Normalization

Batch normalization allows us to employ considerably greater learning rates while being less conscientious about initialization. It is also mentioned that Batch Normalization has a regularization function and can eliminate the need for Dropout in some cases. Because the outputs of the previous layer are influenced by changes in the parameters in the previous layer, the distribution of inputs of each layer varies during the learning process. This is referred to as internal covariate shift. The internal covariate shift is known to slow down learning speed because we have to set a lower learning rate when such sift grows big. In addition, proper parameter initialization is required. This makes training models with saturating nonlinearities infamously difficult. We must lower the internal covariate shift in order to increase learning. We can increase learning speed by maintaining the input distribution of each layer constant throughout the learning process. Whitening the inputs of each layer makes the changes in the inputs of each layer uniform and can reduce the negative impacts of the internal covariate shift. Nevertheless, whitening the inputs of each layer is costly since the whitening requires calculating the covariance matrix of the inputs and solving the eigenvalue problem of the covariance matrix. Instead of whitening using the covariance matrix of the whole training set, Batch Normalizing conducts dimension-wise normalization on the training samples in the mini-batch.

2.3.3 Adam optimizer

The Adam optimizer is a technique for effective stochastic optimization which only involves first-order gradients and consumes little memory. The approach computes individual adaptive learning rates for distinct parameters based on estimations of the first and second moments of the gradients; the name Adam is derived from adaptive moment estimation. The Adam optimizer is being developed to combine the benefits of

two recently popular methods: ADAGRAD, which works well with sparse gradients, and RMSProp [37], which works well in online and non-stationary settings. Adam has several benefits, including the fact that the magnitudes of parameter updates are invariant to gradient rescaling, that its step sizes are roughly constrained by the step size hyperparameter, that it does not need a stationary goal, that it works with sparse gradients, and that it naturally conducts a type of step size annealing.

2.4 Existing solutions

Positioning systems became widely implemented in a vast range of industries, the main goal of this technology is to estimate user location, and navigation-based services rely on it. The most famous example of this system is GPS. However, in an indoor environment, it cannot provide a satisfying positioning performance. Meanwhile, wireless solutions like WiFi, Locata, and Bluetooth offer a satisfying outcome they still do not meet tremendously growing requirements. In comparison with these technologies LiFi has shown to be more accurate results in terms of a positioning error, for instance: for WiFi, it is 1-7m, for Bluetooth 2-5m, while LiFi reaches 0.1-0.35 m [38].

Currently and during the last few years, LiFi has been a hot topic. Many investigations were related to positioning and orientation issues since device orientation is a crucial factor for indoor positioning systems. Among positioning metrics, the best outcome offers RSS, TOA, AOA [39].

In the RSS estimation of position can be detected by a signal received a power which is fit channel model. AOA calculates the angle at the receiver and hence location can be obtained. Finally, TOA characterized by time measuring between signals was sent from transmitter to receiver.

Despite the fact that for the last few years, a lot of research were devoted to LiFi indoor positioning systems, most of them are not suitable for real-life cases. For example, in [40] AOA and RSS techniques were used for single transmitters and multiple optical receivers, and in this solution, the angles between receivers are given. Moreover in [41] the angles of both user equipment receiver and LED transmitter are known as 90 degrees in relation to the ceiling, also the parameters of the room are established. In the physical realm user device is not perfectly aligned to access points and the angles and parameters are usually unknown. In addition, the devices mostly are not in a stable

position and users hold them in the most convenient position and tend to move them. In [42] was proposed a solution that uses tilt angle measuring to define UE position with help of IMU. the disadvantage of this technique is that the accuracy cannot be guaranteed. The mentioned publications were focused on UE positions while UE orientation is a crucial moment. This is a result of using non-linear metrics like RSS and it creates an optimization problem. However, the correct estimation of uncertain orientation parameters is an important aspect that leads to BER reduction. In [43] the uncertain permission power of orientation and position of user devices were investigated, SPAO localization algorithms were proposed as a solution to this non-convex problem based on RSS metrics. Unfortunately, this approach is not realistic for life utilization since it requires six known parameters for the same number of unknown ones.

Another drawback that was identified in the previous solutions is the lack of NLOS consideration in estimating both UE orientation and UE position and count it as a performance diminishing factor. The main reason for this circumstance is the complexity of channel gain in relation to the device orientation and as a result, since the influence of NLOS propagation is not fully comprehended on the LiFi performance it cannot be addressed directly in the optimization approach. In [44] by using Fisher's information analysis the boundaries of performance of light-based systems in NLOS environments are investigated and showed that systems can gain knowledge about user device location through the NLOS channel. Moreover, it was shown that NLOS can lead to improvements for indoor LiFi position systems, however, the way of implying NLOS component for light-based indoor environments was not covered. [45] research answered this question by using deep learning techniques. The main idea is to create offline and online phases. In the first one, the collecting dataset and mapping it ANN models is processed while online is testing these models. This thesis is based on the [46] research and as a contribution, ML algorithms were used to reach high-level analysis.

3 PROCEDURE

In this chapter the whole procedure is described started with room equipment models, expressions that was implicated for data generation, solution parameters and ANN models.

3.1 System and channel model

The work in [33] proposed a system model that was used for this research. The parameters contain room measurements H, W, and L; down-faced Access Points located on the ceiling, where photodiodes are used for data acquisition while LED is responsible for illumination and data transmission. User devices are located randomly and utilize IR-LED (Infrared LED) and PD (Photodiodes). This system supports downlink and uplink at the same time and does not lead to interference and communication between access points and user devices is bidirectional.

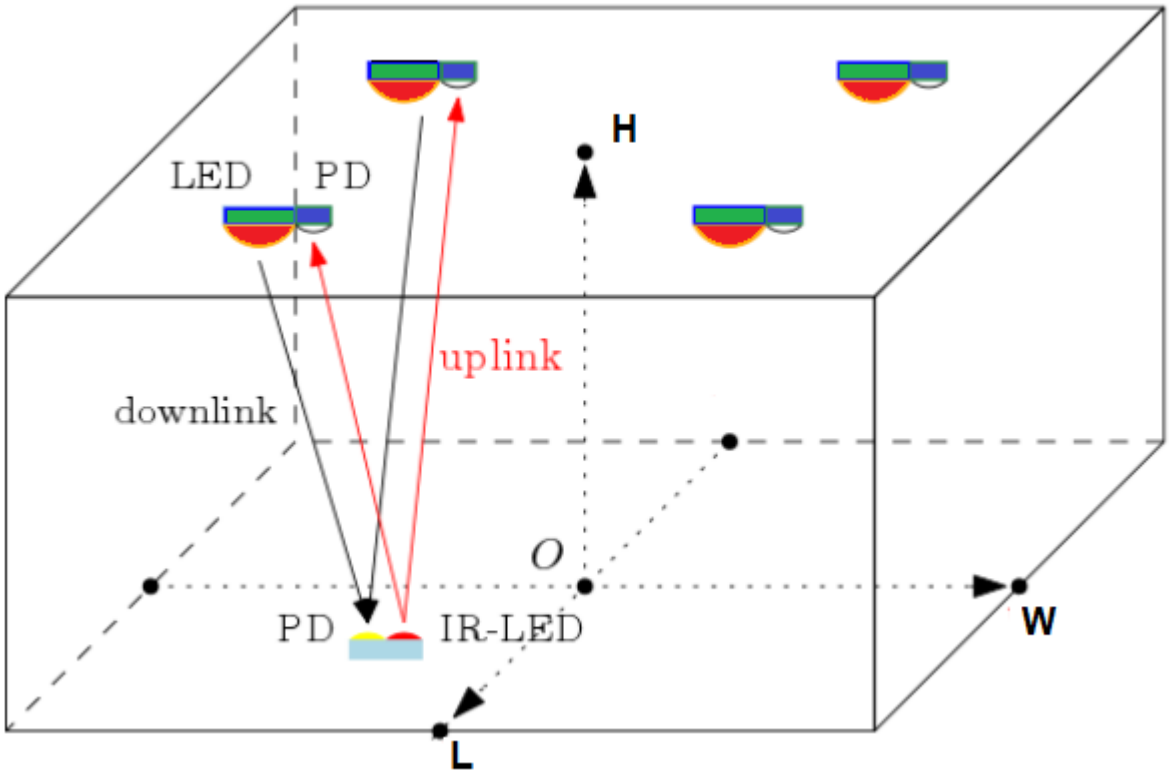

Figure 9: Indoor Environment with established LiFi network [33]

For the signal model the next expression is $y = \lambda Hx + n$ chosen:

$\lambda = TR_p\eta$ where (T: trans-impedance amplifier of gain, R_p – PDs responsivity of APs, η – User Equipment IR-LED current to power conversion efficiency), n – $[n_1, \ldots, n_{Nr}]^T$ – noise vector on at the photodiodes on the access points, H - channel matrix:

$$\mathbf{H} = \begin{pmatrix} h_{1,1} & \cdots & h_{1,N_t} \\ \vdots & \ddots & \vdots \\ h_{N_r,1} & \cdots & h_{N_r,N_t} \end{pmatrix}$$

3.2 Data Generation

The Golden Rule is that there should be no surprises here and this section should be kept fairly short. Present the conclusions one by one in a logical order. each should be brief and self-contained. Each conclusion must be drawn in a logical order from what has gone before: *fact, fact, fact* therefore *conclusion* based on your analysis and discussion.

Following we describe the steps of data collection:

- A collection of 3D positions (x, y,) is built. Assuming randomly located inside the observed environment user device that communicates with access points on the ceiling. Samples are generated by using the next mechanism: for each random variable x, the PDF is defined by f_x. Moreover, the probability density function for user device 3D position is denoted by:

$$f_x(x) = \frac{1}{L}\mathcal{U}_{[-\frac{L}{2},\frac{L}{2}]}(x),$$
$$f_y(y) = \frac{1}{W}\mathcal{U}_{[-\frac{W}{2},\frac{W}{2}]}(y),$$
$$f_z(z) = \frac{1}{H_{\text{device}}}\mathcal{U}_{[0,H_{\text{device}}]}(z),$$

- Considering Ω as a movement direction angle, uniformly build a sample from [0, 360] while user equipment is in static or changing position. Calculations have been done from the East direction.
- Generating three orientation angles. By using data from experiments [44,45] the orientation information of yaw α, pitch β, roll γ are measured and recorded. The final output build utilizing truncated Laplace distribution and specification like standard and mean deviation (μ_α, σ_α) for α, (μ_β, σ_β) for β and (μ_γ, σ_γ) for γ.

	α	β	γ
Mean	Ω-90	40.78	-0.84
Standard deviation	3.67	2.39	2.21

Figure 10: Mean and Standard Deviation for rotation angles

[33]

```
mu_alpha = omega-pi/2;
alpha0 = laprnd(mu_alpha,sigma_alpha,1,1);
alpha = 1/2*(1-sign(alpha0))*(2*pi-abs(alpha0)) + 1/2*(1+sign(alpha0))*alpha0;
beta = laprnd(mu_beta,sigma_beta,1,1);
gamma = laprnd(mu_gamma,sigma_gamma,1,1);
UE_orientation = [alpha;beta;gamma];
```

- The computation of the resulting H-matrix.

Channel gain h both of NLOS and LOS calculations described as:

$$h_{i,j} = h_{i,j}^{LOS} + h_{i,j}^{NLOS},$$

where $i \in [1, N_r]$ – PD of AP, $j \in [1, N_t]$ - IR-LED. The values of each $h_{i,j}$ are based on the user equipment IR-LED orientation and corresponding access point location. Hence by using expressions in [47-48-49] the channel gain for LOS and NLOS are illustrated below:

$$h_{ij_{\mathrm{LOS,Gen.Lamb.}}} = \begin{cases} \dfrac{(m+1)A}{2\pi d^2}\cos^m\phi T_s(\psi)g(\psi)\cos\psi, & 0 \le \psi \le \Psi_c \\ 0, & \theta > \Psi_c. \end{cases}$$

$$h_{i,j}^{\mathrm{NLOS}} = \mathbf{r}^{\mathrm{T}}\mathbf{G}_\zeta(\mathbf{I} - \mathbf{E}\mathbf{G}_\zeta)^{-1}\mathbf{t},$$

where the vectors t and r denote the LOS link between the jth IR-LEDs and all of the room's surface elements, and from all of the room's surface elements to the ith AP, respectively. The reflectivity matrix of all K reflectors is G = diag(1,...,K); E is the LOS transfer function of size K for the linkages between all surface elements, and IK is the unity matrix of order K.

```
H_LOS(i) = LOS_channel_gain_Single_LED(UE_position,UE_orientation,PD_position,Psi_Fov,Phi_Fov,H0,m);
H_NLOS(i) = NLOS_channel_gain_Single_LED(UE_position,UE_orientation,PD_position,dimension,Psi_Fov,Phi_Fov,rho,H0,m);
H_total(i) = H_LOS(i) + H_NLOS(i);
```

- a power of random electrical emission P_{elec} is created uniformly between [0, P^{max}_{elec}], where P^{max}_{elec} is the UE's maximum possible electrical emission power.
- By applying the next expression with using calculations above the corresponding SNR vector is computed:

$$\rho_i = \frac{\left(\lambda \sum_{j=1}^{N_t} h_{i,j}\right)^2 P_{\mathrm{elec}}}{\sigma_n^2},$$

- Finally, the resulting SNR vector is recorded as a feature vector in the dataset, together with the matching 3Dposition and orientation angles (x, y, z, α, β, γ) as a label vector [46].

3.3 Learning Models

Machine Learning techniques, especially artificial neural network algorithms are the most suitable solution for nonlinear and complicated optimization problems in wireless communication. Generally, it is applied in the physical layer. In this thesis, the goal is to define a correct g(*P*; ·), parametric mapping between created features and labels, between received SNR vector and user device position and orientation respectively. This can be achieved by picking the optimal set of *P* that yields the best mapping for a given EE metric.

$$(x, y, z, \alpha, \beta, \gamma) = g\left(\mathcal{P}^*, \boldsymbol{\rho}\right)$$

Four models were chosen: KNN and SVM, CNN, and MLP. Since the three last

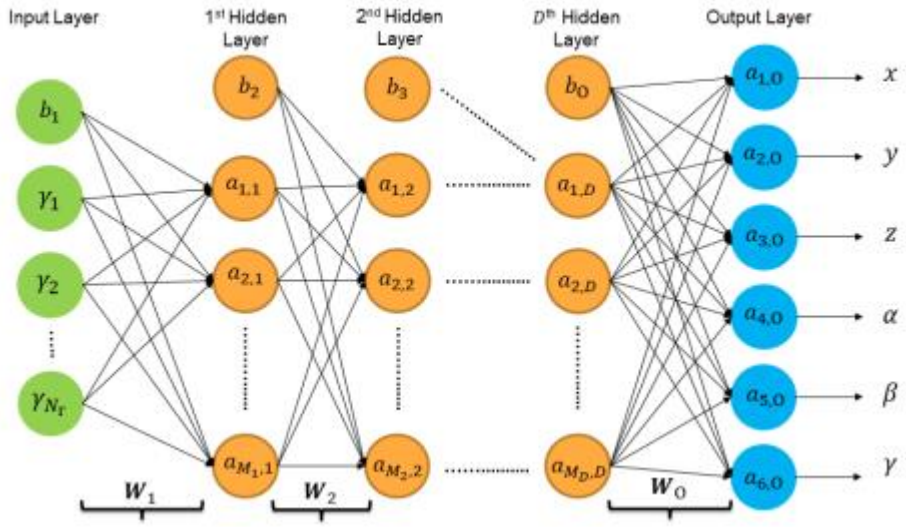

Figure 11: ANN architecture

algorithms were modified to be presented as ANN, the architecture is developed from the multiple layers: input, D-hidden, and output layers where D – depth of NN. Each layer consists of artificial neurons and their connections [47].

In the next steps each layer is described:

1. Input layer: the SNR vector that represents feature b bias is given to NN as input data
2. D-hidden layer: this layer includes M- artificial neurons and their relation. Each artificial neuron can perform a mathematical operation on its inputs before implementing an activation function to generate a signal that is being sent to the next layer. The expression describes the propagation

$$v_{i,j} = a_{i,j}\left[\mathrm{t}\left(\mathbf{u}_{j-1}, \mathbf{w}_{i,j}, b_{i,j}\right)\right]$$

where j∈[1,D], i∈[1,M_j], $b_{i,j}$ – scalar bias, and the a is activation function, u – input, w – vector of weights, and t linear transformation that based on the type of chosen artificial neural network and bias.

3. Output layer. This part is composed of six neurons and each of them is held to account for the estimation of parameters in (x, y, z, α, β, γ). Following the logic that this layer is (D+1), the output vector is denoted as:

$$u_j = v_j = [v_{1,j}, v_{2,j}, \ldots, v_{Mj,j}]^T$$

While propagation rules defined in this level like:

$$\begin{aligned} x &= a_{1,\mathrm{o}}\left[\mathrm{t}\left(\mathbf{u}_D, \mathbf{w}_{1,\mathrm{o}}, b_{i,\mathrm{O}}\right)\right], \qquad \alpha = a_{4,\mathrm{o}}\left[\mathrm{t}\left(\mathbf{u}_D, \mathbf{w}_{4,\mathrm{o}}, b_{i,\mathrm{O}}\right)\right], \\ y &= a_{2,\mathrm{o}}\left[\mathrm{t}\left(\mathbf{u}_D, \mathbf{w}_{2,\mathrm{o}}, b_{i,\mathrm{O}}\right)\right], \qquad \beta = a_{5,\mathrm{o}}\left[\mathrm{t}\left(\mathbf{u}_D, \mathbf{w}_{5,\mathrm{o}}, b_{i,\mathrm{O}}\right)\right], \\ z &= a_{3,\mathrm{o}}\left[\mathrm{t}\left(\mathbf{u}_D, \mathbf{w}_{3,\mathrm{o}}, b_{i,\mathrm{O}}\right)\right], \qquad \gamma = a_{6,\mathrm{o}}\left[\mathrm{t}\left(\mathbf{u}_D, \mathbf{w}_{6,\mathrm{o}}, b_{i,\mathrm{O}}\right)\right], \end{aligned}$$

Where the weights vector $w_{k,0}$, activation function $a_{k,O}$ of the k-neuron.

The choice of a linear transformation is based on the type of utilized algorithm. For CNN the 2D transformation was employed, while activation functions were set as Linear and rectified linear unit, RELU, which is the default way to solve non-linear optimization problems. By using convolutional operator ⊛ the transformation is expressed as:

$$\begin{cases} t\left(u_{j-1}, w_{i,j}, b_j\right) = w_{i,j} \circledast u_{j-1} + b_j \\ t\left(u_D, w_{k,o}, b_O\right) = w_{k,o} \circledast u_D + b_{i,O}, \end{cases}$$

Where j ∈ [1, D], i ∈ [1, M_j] and k ∈ [1, 3].

On the other hand, for MLP was used the weighted sum as *t* with the same linear ReLu activation function.

The expression is defined as:

$$\begin{cases} t\left(u_{j-1}, w_{i,j}, b_j\right) = {w_{i,j}}^T u_{j-1} + b_j \\ t\left(u_D, w_{k,o}, b_O\right) = {w_{k,o}}^T u_D + b_{i,O}, \end{cases}$$

KNN algorithm was chosen as the one of typical algorithms that gave reasonable results for VLC indoor positioning according to [48,49] for comparing results. The architecture does not use Ann and denotes standard code.

Support Vector Machine is one of machine learning method. RBF kernel is a popular kernel function used in various kernel learning algorithm. In particular, it is commonly used in SVM [50]. In this paper, the SVM is used to localize the position of AP and classify it. The input data which is used are RSS and the vector location of each AP. The kernel function which is used in the localization is RBF (radial basis function). The RBF calculation is showed in equation

$$K(x, x') = \exp\left(\frac{\|x-x'\|^2}{2\sigma^2}\right)$$

Here, $\|x - x'\|^2$ may be recognized as the squared Euclidean distance between the two feature vectors. σ is the variance and our hyperparameter.

3.4 Training process

Once the selected Ann models are set the next stage is to train the model. To map between feature and label, received SNR vector and UD orientation angle and 3D-position the correct obtaining parameter set is required. The following expression is characterizing this set: P = W ∪ B, where W = {W_O, W_j |j ∈ [1, D]}, such that W_O = [$w_{j,O}$, $w_{2,O}$, ..., $w_{MO,O}$], and for all j ∈ [1, D], W = $w_{1,j}$, $w_{2,j}$, ..., $w_{Mj,j}$, and B = {b_O, b_j |j ∈ [1, D]}, such that b_O = [$b_{j,O}$, $b_{2,O}$, ..., $b_{MO,O}$] ^{T} , and for all j ∈ [1, D], b_j = [$b_{1,j}$, $b_{2,j}$, ..., $b_{Mj,j}$]^T

In this thesis, the mean square error (MSE) loss function was utilized in metrics MSE and MAE as a solution to the regression problem and was optimal parameter set *P** was obtained according to [33] in the way as:

$$\mathcal{P}^* = \underset{\mathcal{P}}{\operatorname{argmin}}\ \mathrm{L}_2(\mathcal{P}), = \underset{\mathcal{P}}{\operatorname{argmin}}\ \frac{1}{N_{\text{train}}} \sum_{l=1}^{N_{\text{train}}} ||\mathbf{P}_l - \widehat{\mathbf{P}}_l\,(\mathcal{P}, \boldsymbol{\rho}_l)\,||_2^2,$$

where L_2 – selected MSE loss, N_{train} – the dataset size (N=3, N = 5, N = 6), $\hat{P}_l$ is derived from the selected Ann concerning the set of parameters P by estimating label vectors associated with the lth feature vector l of the dataset, respectively. The gradient descent approach can be used to solve the optimization problem. In reality, by iteratively advancing in the direction of steepest descent, as defined by the negative of the gradient, gradient descent can be utilized to minimize the loss function MSE. Hence standard Adam optimizer was implicated for each utilized ANN:

```
optimizer = Adam(learning_rate=0.001, beta_1=0.9, beta_2=0.999, amsgrad=False)
model.compile(optimizer=optimizer, loss='mse',  metrics=['mae'])
```

3.5 Testing phase

The evaluation of new unseen data, in this case, a new vector of received SNR was developed in the given chapter. The testing criteria were chosen according to [50] and can be denoted as:

- The average EE (estimation error): illustrates the average difference between the correct label vector and the estimated one.
- Precision: it denotes an EE that exceeds 90% of the total number of possible EE.
- The computational time: it is the average time required to estimate the label vector of a certain feature vector during a single online prediction session [51].

3.6 Simulation parameters

This section contains parameters that were used to run the simulation. Physical parameters were chosen based on the previous works [33,51]. Table 1 and Table 2 consider physical parameters and Ann specifications respectively. Two different typical rooms were considered with dimensions 5m×5m×3m and 7m×7m×3m equipped with 16 AP placed downwards on the ceiling. Standard smartphones are used as UD, one IR-LED placed on the screen. UE has a random location.

For an accurate result, the size of the dataset was set as N^6. Each Ann technique consists of an input layer, four hidden layers, and an output layer. Adam optimizer was used and kernel size equals 16 parameters. For SVM and CNN the number of filters is 32 and 64 respectively, and 256 neurons for MLP.

Table 2: Parameters for simulation [33]

Name of the Parameter	Symbol	Value
Number of AP	N_r	16
Dimensions of the room 1	L × W × H	5m×5m×3m
Dimensions of the room 2	L × W × H	7m×7m×3m
LED half-power semi angle	$\Phi_{1/2}$	60◦
PD responsivity	R_p	0.6 A/W
PD geometric area	A_g	1 cm2
Optical concentrator refractive index	n_c	1
Maximum UE's height	H_{device}	1.5m
Maximum UE's power	P^{max}_{elec}	0.01 W

Reflection coefficient of the walls	ζ	0.7
Field of view of the IR-LEDs	Φ	90◦
Field of view of the PDs	Ψ	90◦
System Bandwidth	B	10 MHz
Noise power spectral density	N_0	10^{-21} W/Hz

Table 3: ANN specifications

Epoch	30
Kernel for SVM	Linear, rbf
Trained dataset size №1	N6
Number of D hidden layers or depth of utilized Ann	4
Number of filters for SVM and CNN	64
Number of neurons for MLP	256
Kernel size for CNN and MLP	16
Optimizer for Ann	Adam
Partition (test, train)	(0.9, 0.1) * N

The architecture of hidden layers for SVM, MLP, CNN is described in Figure 12.

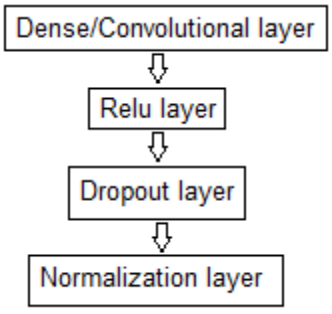

Figure 12: Hidden Layers

Each input and output are linked to each other by weight is the dense layer in MLP. Moreover, it is worth mentioning that it is a completely connected layer. The CNN and SVM use a convolutional layer that uses convolution between filter and kernel. On the second layer, the RELU activation function is implicated. As a next step, the prevention of Ann overfitting a dropout layer reduces a network with a certain probability. Batch

Normalization achieves fast and smooth and accurate training, by standardizing input, hence the output is supposed to have a value near zero mean and unit variance. On the other hand, chosen KNN has only a Dense and linear activation layer which is set a connection between input and output [33, 52].

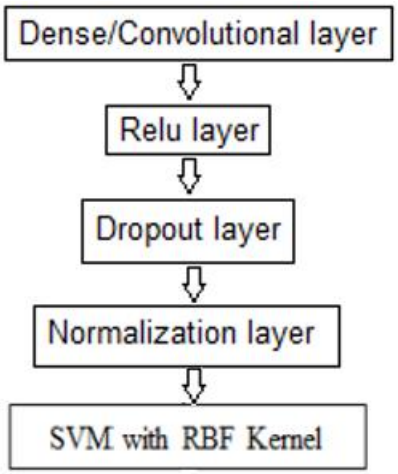

Figure 13: SVM architecture

The output is calculated by multiplying RSSI and the random input weight. The activated function is obtained using Relu function. Estimated location is calculated with multiplying vector y[j] and output weight.

4 RESULTS

4.1 Evaluation of learning models and performance estimation

Three different ANN models and KNN models were tested on generated for 2 rooms different datasets which were grouped into batches. Each epoch represents a single pass through the complete training set. Figures 7, 8, 9 represent a mse loss against epoch for NLOS+LOS generated dataset. For CNN, MLP and SVM total of 30 numbers of the epoch were utilized for the training phase. Each graph illustrates the absence of overfitting, which means that each model is able to generalize new unseen data. Moreover, both types of losses decline with incrementing epoch number

In the employed ANN models, each hidden layer includes a dropout layer, which is a fundamental stage to prevent outfitting. Dropout regularization drop each individual neuron with certain probability which leads to reduction of ANN network. Figures 14, 15, 16 show the number of trainable and non-trainable parameters for each model.

```
_________________________
Total params: 798,342
Trainable params: 798,086
Non-trainable params: 256
```

Figure 14: SVM number of parameters

```
_________________________
Total params: 205,062
Trainable params: 204,550
Non-trainable params: 512
```

Figure 15: CNN number of parameters

```
Total params: 54,534
Trainable params: 53,510
Non-trainable params: 1,024
```

Figure 16: MLP number of parameters

Figures 17,18,19 show the training and validation losses of the MLP, SVM and CNN models assessed in terms of mean squared-error (MSE), versus the epoch index for the two datasets investigated. In particular, the data proposed for training ANN models is split into two subsets: one for training the models to obtain the weights P, and the other for confirming the generalization error of the obtained weights on unknown data. As a result, each epoch represents a single pass through the full training set. Moreover,

training models with higher number of epochs performs better than 15 epochs. The training and validation losses decrease as the epoch index grows, as shown in Figures 17,18,19, indicating that the resulting ANN models are not overfitting and can generalize well over unknown data during online phase.

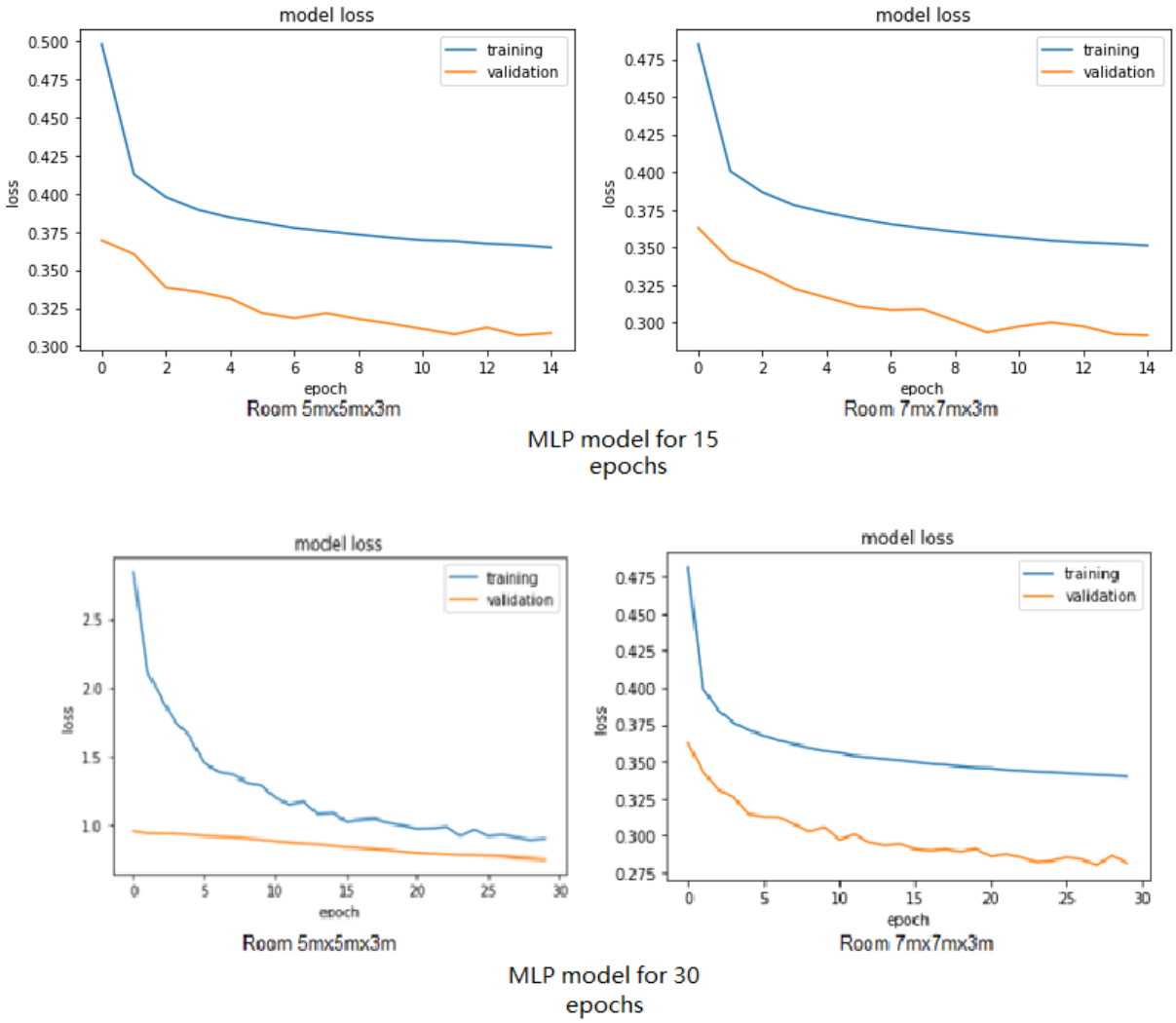

Figure 17 MLP model

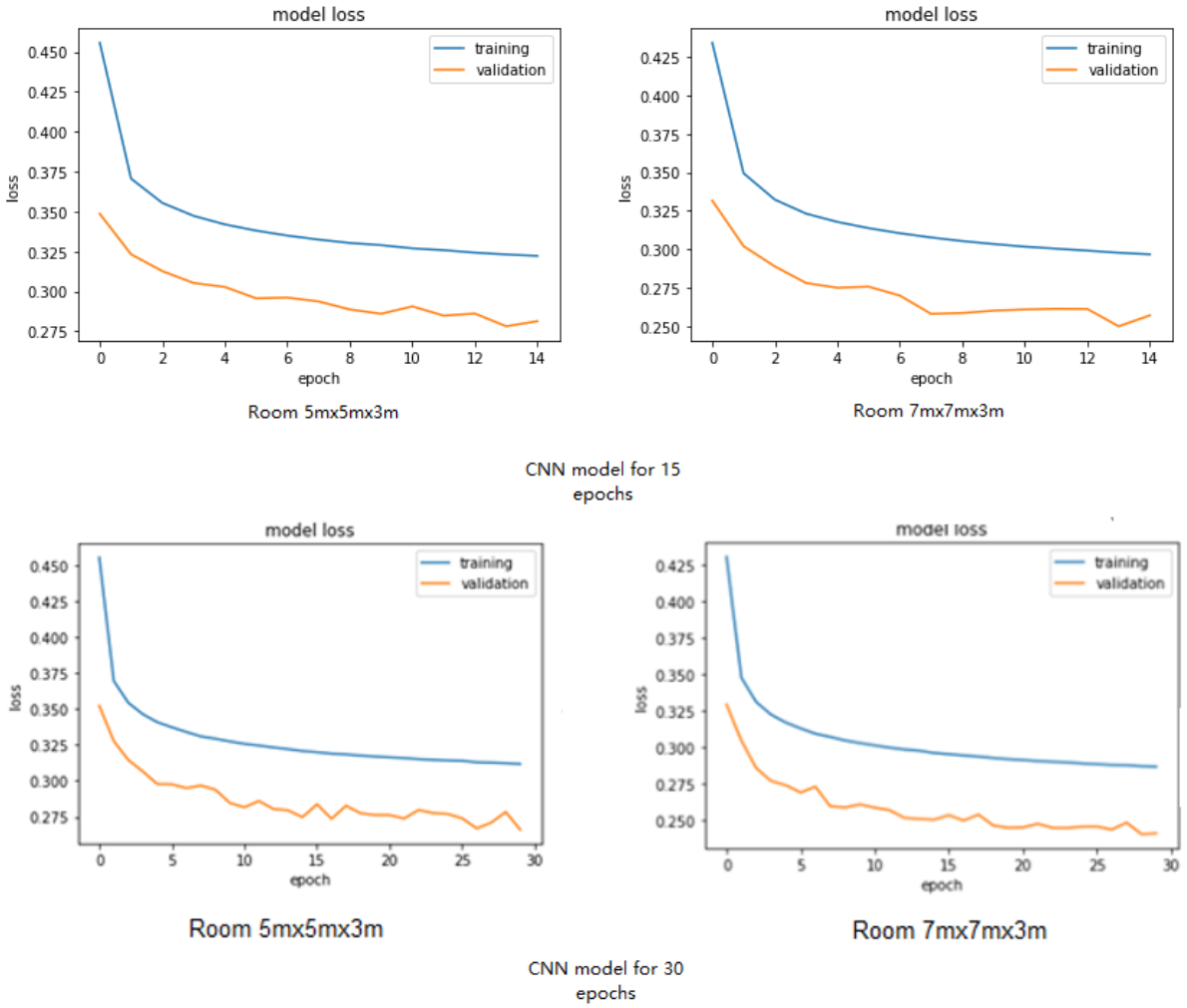

Figure 18 CNN model

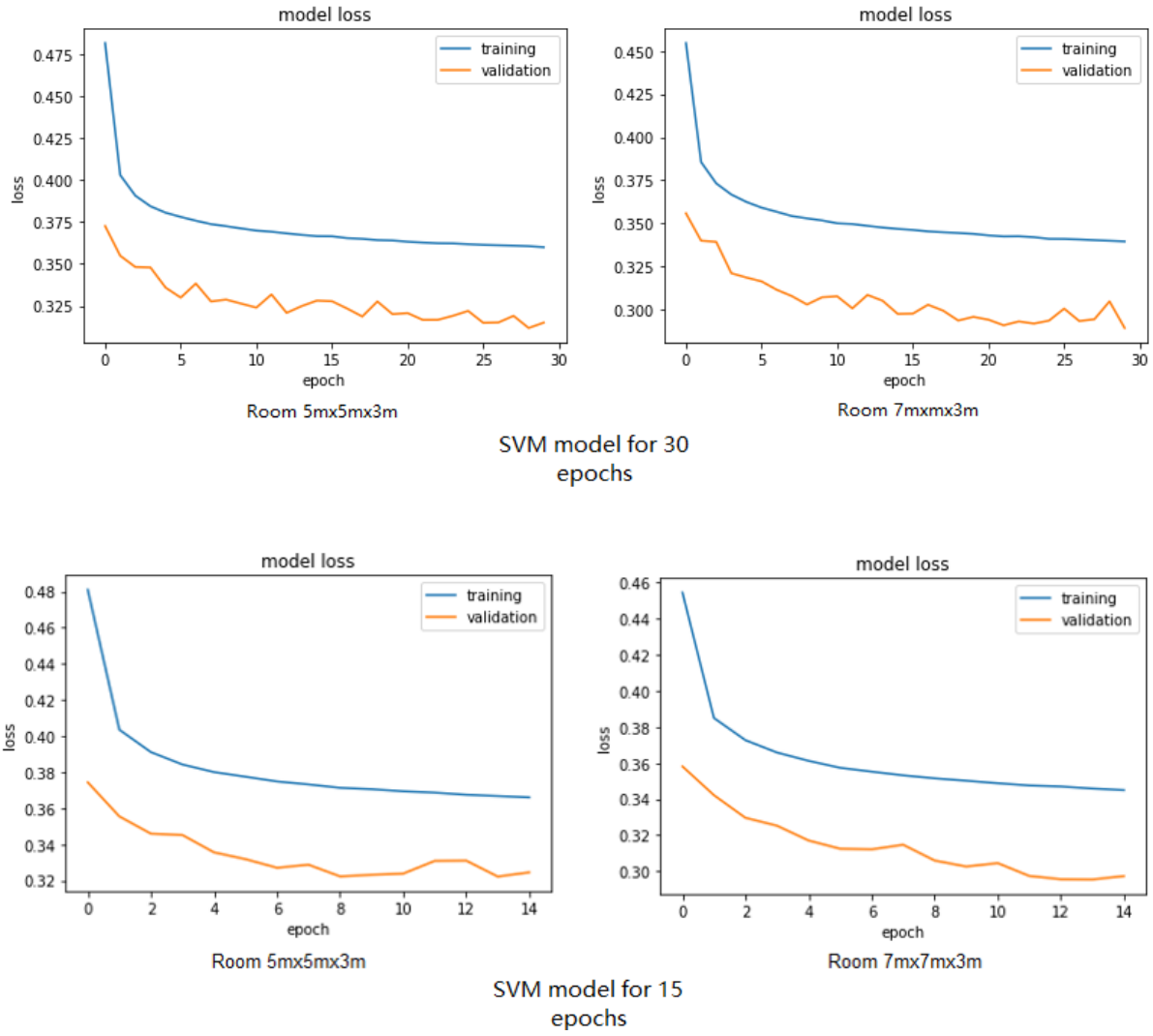

Figure 19 SVM model

In total Figures 17, 18 and 19 demonstrate four cases for each model:

- Dataset N^5
- Dataset N^6
- Epochs number 30
- Epochs number 15

Graphs 17,18,19 illustrate that enhancing the dataset size improves the ANN models' learning efficiency. This is mostly because having more data points allows the ANNs to gain a better understanding the random behaviour of the environment, which is expressed in terms of the effects of the UE's random position and orientation on the instantaneous received SNR.

Table 4 Comparison between ANN techniques and KNN in terms of average error in cm and degree

		CNN		SVM		MLP		KNN	
		$5x5x3m^3$	$7x7x3m^3$	$5x5x3m^3$	$7x7x3m^3$	$5x5x3m^3$	$7x7x3m^3$	$5x5x3m^3$	$7x7x3m^3$
Location	x	3.1	2.7	4.10	3.07	7.05	6.08	10.31	11.01
	y	6.01	5.25	3.46	2.03	5.03	8.021	9.03	9.001
	z	5.04	4.94	6.89	7.29	6.73	9.74	8.42	10.92
Orientation	α	7.29	7.27	7.04	5.90	7.70	6.73	7.53	5.70
	β	2.30	2.26	3.18	3.14	3.20	3.15	11.78	11.73
	γ	2.87	2.44	1.44	1.01	1.05	1.047	3.92	3.18

Table 4 illustrates the average estimation error for each predicted parameter set P'(x, y, z, α, β, γ). The data gained by ANN models outperform KNN in each parameter. For location and estimation, CNN and SVM demonstrate higher precision than MLP. However, each CNN technique shows a similar trend for orientation angles, for instance, the α-angle showing the worst results in comparison with β and γ. This phenomenon is a consequence of the fact the last two rotation angles have fixed means while yaw α uses Ω-90^{o} which leads to slight fluctuations. Hence this parameter suffers from higher estimation error.

	loss	mae	val_loss	val_mae	epoch
0	2.473910	1.207828	0.963299	0.801861	0
1	1.921540	1.051481	0.954409	0.797809	1
2	1.708724	0.965632	0.946167	0.791876	2
3	1.351763	0.865712	0.934445	0.788033	3
4	1.336540	0.855027	0.931489	0.787081	4
5	1.264224	0.816453	0.922535	0.781640	5
6	1.190055	0.804528	0.927446	0.784177	6
7	1.212483	0.787962	0.918963	0.778291	7
8	1.103619	0.768318	0.917457	0.777685	8
9	1.129435	0.770881	0.914803	0.776371	9
10	1.090829	0.756926	0.917970	0.776623	10
11	1.062627	0.742678	0.915114	0.775797	11
12	1.019445	0.720598	0.919085	0.777233	12
13	1.033938	0.729115	0.910060	0.772604	13

Figure 20: Loss parameters after each epoch

Figure 20 proofs that, as the epochs increase, the gap between validation and training data is decreasing as well as both types of losses.

4.2 Evaluation of computational time

In terms of computation, MLP shows the fastest results among all applied Ann models. For each epoch step, it takes approximately 40 seconds and 20 minutes in total, while CNN used 12 hours for 30 epochs. On each epoch SVM spent 1.30 hours and 8 hours 18 min only for 15 of the epochs. Figures 21, 22, 23 demonstrate the examples in terms of time consumption.

```
Epoch 15/15
5274/5274 [==============================] - 1352s 256ms/step -
al_mae: 0.2979 - val_mse: 0.3244
Wall time: 8h 18min 58s
```

Figure 21: Wall time for SVM

```
Epoch 30/30
5274/5274 [==============================] - 43s 8ms/step
ae: 0.2669 - val_mse: 0.2814
Wall time: 19min 56s
```

Figure 22: Wall time for MLP

```
Epoch 30/30
5625/5625 [==============================] - 985s 180ms/step
l - val_loss: 0.2412 - val_mae: 0.2318
Wall time: 8h 4min 10s
```

Figure 23: Wall time for CNN

Table 5 Overall computation time for each case

Model	Testing 15 epochs		Testing 30 epochs	
	Dataset N^5	Dataset N^6	Dataset N^5	Dataset N^6
CNN	5 h 3 min	5 h 16 min	8 h 4 min	8 h 16 min
MLP	9 min	10 min	19 min	21 min
SVM	8 h 16 min	8 h 21 min	12 h 8 min	12 h 34 min

4.3 Discussion

Figures 17,18,19 demonstrate clearly that the training and validation losses decrease as the epoch index grows indicating thus that the resulting ANN models are not overfitting and can generalize well over unknown datasets in online testing. Furthermore, the training loss is greater than the validation loss and the reason is dropout utilization. Dropout is used throughout the training phase, like any similar regularization strategy, but not during the validation phase [47]. In other words, at validation time, regularization processes such as dropout are disabled, resulting in a training loss higher than the validation loss.

It is observed that SVM algorithm gives low accuracy compared to CNN. SVM algorithm is generally used to solve classification problems in which the dataset is divided into minimum number of classes and for dataset which is of a lower dimension. SVM uses a linear decision boundary to separate the datapoints into different classes. In SVM it is possible to make the decision boundary soft using slack variable and box constraint. This fails when the number of classes become high and when the datapoints become

skewed or imbalanced, like for the given problem statement. SVM requires the longest time to process neither CNN nor MLP. However, location estimation results for SVM are better than MLP.

Positioning systems using ANN and DNN algorithms is a hot topic today. It can be utilized for different needs such as easing the navigation for blind people for or in case of emergency to localize people to be survived. Despite the fact that systems need developments and improvement it can be used in these days.

5 CONCLUSION

Finally, for indoor location systems using LiFi, this thesis investigated, analysed, developed, and evaluated three ANN models such as SVM, CNN, and MLP, and compared results with the KNN approach by using Python and Keras library. Each model has the goal obtain an efficient mapping between receiver SNR vector and corresponding UE position and orientation. User device was placed randomly. The RSS fingerprinting metric was implicated to generate the dataset. Moreover, all prediction models have used LOS+NLOS components. To achieve high training results the epoch index was utilized. Moreover, the dataset was divided into several batches, and the normalization layer standardize the data.

Additionally, the obtained results were compared with KNN and each other, the superiority of Ann models was illustrated in terms of performance. The differences between computational time and estimation error were considered.

Finally, in this thesis the following tasks have been performed.

- Multiple Ann and ML techniques have been investigated and analyzed.
- The modern knowledge about LiFi approaches have been utilized.
- SVM, CNN, and MLP models have been chosen to represent ANN.
- KNN has been used for comparison with deep learning techniques.
- The supremacy of Ann models was illustrated.
- The dataset generation was analyzed and developed for 2 rooms in size of N^5
- Different channel matrixes have been decompounded and the modern ones have been applied.

ABBREVIATIONS - ACRONYMS

ANN	*Artificial Neural Network*
AOA	*Angle of Arrival*
AP	*Access Point*
CNN	*Convolutional Neural Network*
DD	*Direct Detection*
IM	*Intensity Modulation*
EE	*Estimation Error*
IR-LED	*Infrared Light Emitting Diode*
KNN	*K-nearest Neighbours Algorithm*
LED	*Light Emitting Diode*
LiFi	*Light Fidelity*
LOS	*Line of sight*
MAE	*Mean absolute error*
ML	*Machine learning*
MLP	*Multilayer perceptron*
MSE	*Mean square error*
NLOS	*Non-Line of Sight*
PD	*Photodiode*
PDF	*Probability Density Function*
RELU	*Rectified Linear Unit*
RSS	*Received signal strength*
SNR	*Signal-to-noise ratio*
SVM	*Support Vector Machine*
TOA	*Time of Arrival*
UD	*User Device*
UE	*User Equipment*

REFERENCES

[1] J. G. Andrews *et al.*, 'What Will 5G Be?', *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014, doi: 10.1109/JSAC.2014.2328098.

[2] A. Yassin *et al*., "Recent Advances in Indoor Localization: A Survey on Theoretical Approaches and Applications," in *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 1327-1346, Secondquarter 2017

[3] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, 'Deep Learning for Wireless Physical Layer: Opportunities and Challenges', *arXiv:1710.05312 [cs, math]*, Oct. 2017, Accessed: May 12, 2022. [Online]. Available: http://arxiv.org/abs/1710.05312

[4] M. Dehghani Soltani, A. A. Purwita, I. Tavakkolnia, H. Haas, and M. Safari, 'Impact of Device Orientation on Error Performance of LiFi Systems', *IEEE Access*, vol. 7, pp. 41690–41701, 2019, doi: 10.1109/ACCESS.2019.2907463.

[5] Y. Chen, W. Guan, J. Li and H. Song, "Indoor Real-Time 3-D Visible Light Positioning System Using Fingerprinting and Extreme Learning Machine," in *IEEE Access*, vol. 8, pp. 13875-13886, 2020, doi: 10.1109/ACCESS.2019.2961939.

[6] H. Q. Tran and C. Ha, "High Precision Weighted Optimum K-Nearest Neighbors Algorithm for Indoor Visible Light Positioning Applications," in *IEEE Access*, vol. 8, pp. 114597-114607, 2020, doi: 10.1109/ACCESS.2020.3003977.

[7] Lan, Haiyu, Chunyang Yu, Yuan Zhuang, You Li, and Naser El-Sheimy. 2015. "A Novel Kalman Filter with State Constraint Approach for the Integration of Multiple Pedestrian Navigation Systems" *Micromachines* 6, no. 7: 926-952. https://doi.org/10.3390/mi6070926.

[8] Y. Zhuang and N. El-Sheimy, "Tightly-Coupled Integration of WiFi and MEMS Sensors on Handheld Devices for Indoor Pedestrian Navigation," in *IEEE Sensors Journal*, vol. 16, no. 1, pp. 224-234, Jan.1, 2016, doi: 10.1109/JSEN.2015.2477444.

[9] Y. Zhuang, Z. Syed, Y. Li and N. El-Sheimy, "Evaluation of Two WiFi Positioning Systems Based on Autonomous Crowdsourcing of Handheld Devices for Indoor Navigation," in *IEEE Transactions on Mobile Computing*, vol. 15, no. 8, pp. 1982-1995, 1 Aug. 2016, doi: 10.1109/TMC.2015.2451641.

[10] Zhuang, Y., You Li, Haiyu Lan and Naser El-Sheimy. "Design and Evaluation of an Improved Integration of WiFi Fingerprinting and MEMS Sensors in Smartphones." (2015).

[11] P. Yang and W. Wu, "Efficient Particle Filter Localization Algorithm in Dense Passive RFID Tag Environment," in *IEEE Transactions on Industrial Electronics*, vol. 61, no. 10, pp. 5641-5651, Oct. 2014, doi: 10.1109/TIE.2014.2301737.

[12] S. -H. Fang, C. -H. Wang, T. -Y. Huang, C. -H. Yang and Y. -S. Chen, "An Enhanced ZigBee Indoor Positioning System With an Ensemble Approach," in *IEEE Communications Letters*, vol. 16, no. 4, pp. 564-567, April 2012, doi: 10.1109/LCOMM.2012.022112.120131.

[13] F. Yao, A. Keller, M. Ahmad, B. Ahmad, R. Harrison and A. W. Colombo, "Optimizing the Scheduling of Autonomous Guided Vehicle in a Manufacturing Process," *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, 2018, pp. 264-269, doi: 10.1109/INDIN.2018.8471979.

[14] Eppner, C., Höfer, S., Jonschkowski, R. *et al.* Four aspects of building robotic systems: lessons from the Amazon Picking Challenge 2015. *Auton Robot* **42,** 1459–1475 (2018). https://doi.org/10.1007/s10514-018-9761-2

[15] B. Zhou and Q. Chen, "On the Particle-Assisted Stochastic Search Mechanism in Wireless Cooperative Localization," in *IEEE Transactions on Wireless Communications*, vol. 15, no. 7, pp. 4765-4777, July 2016, doi: 10.1109/TWC.2016.2545665.

[16] H. Haas, L. Yin, Y. Wang and C. Chen, "What is LiFi?," in *Journal of Lightwave Technology*, vol. 34, no. 6, pp. 1533-1544, 15 March15, 2016, doi: 10.1109/JLT.2015.2510021.

[17] I. Tavakkolnia, C. Chen, R. Bian and H. Haas, "Energy-Efficient Adaptive MIMO-VLC Technique for Indoor LiFi Applications," *2018 25th International Conference on Telecommunications (ICT)*, 2018, pp. 331-335, doi: 10.1109/ICT.2018.8464933.

[18] M. A. Arfaoui *et al*., "Physical Layer Security for Visible Light Communication Systems: A Survey," in *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1887-1908, thirdquarter 2020, doi: 10.1109/COMST.2020.2988615.

[19] Qualcomm, “Lumicast lights the way to a more personal retail experience,” [Online]. Available: https://www.qualcomm.com/news/onq/2017/ 03/20/lumicast-lights-way-more-personal-retail-experience, Jun. 2018.

[20] Z. Zeng, M. D. Soltani, H. Haas and M. Safari, "Orientation Model of Mobile Device for Indoor VLC and Millimetre Wave Systems," *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, 2018, pp. 1-6, doi: 10.1109/VTCFall.2018.8691024.

[21] M. D. Soltani, “Analysis of Random Orientation and User Mobility in LiFi Networks,” The University of Edinburgh, Jul. 2019, [Online], Available: https://era.ed.ac.uk/handle/1842/35965

[22] Naveed Ul Hassan, Aqsa Naeem, Muhammad Adeel Pasha, Tariq Jadoon, and Chau Yuen. 2015. Indoor Positioning Using Visible LED Lights: A Survey. ACM Comput. Surv. 48, 2, Article 20 (November 2015), 32 pages. https://doi.org/10.1145/2835376

[23] S. Yang, H. Kim, Y. Son and S. Han, "Three-Dimensional Visible Light Indoor Localization Using AOA and RSS With Multiple Optical Receivers," in *Journal of Lightwave Technology*, vol. 32, no. 14, pp. 2480-2485, 1 July15, 2014, doi: 10.1109/JLT.2014.2327623.

[24] H. Sharifi, A. Kumar, F. Alam and K. M. Arif, "Indoor localization of mobile robot with visible light communication," *2016 12th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, 2016, pp. 1-6, doi: 10.1109/MESA.2016.7587166.

[25] W. Zhang, M. I. S. Chowdhury, and M. Kavehrad, ‘Asynchronous indoor positioning system based on

visible light communications', *OE*, vol. 53, no. 4, p. 045105, Apr. 2014, doi: 10.1117/1.OE.53.4.045105.

[26] Z. Zhou, M. Kavehrad, and P. Deng, 'Indoor positioning algorithm using light-emitting diode visible light communications', *OE*, vol. 51, no. 8, p. 085009, Aug. 2012, doi: 10.1117/1.OE.51.8.085009.

[27] L. Yin, X. Wu and H. Haas, "Indoor Visible Light Positioning with Angle Diversity Transmitter," *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)*, 2015, pp. 1-5, doi: 10.1109/VTCFall.2015.7390984.

[28] K. Qiu, F. Zhang and M. Liu, "Visible Light Communication-based indoor localization using Gaussian Process," *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 3125-3130, doi: 10.1109/IROS.2015.7353809.

[29] M. D. Soltani, A. A. Purwita, Z. Zeng, H. Haas and M. Safari, "Modeling the Random Orientation of Mobile Devices: Measurement, Analysis and LiFi Use Case," in *IEEE Transactions on Communications*, vol. 67, no. 3, pp. 2157-2172, March 2019, doi: 10.1109/TCOMM.2018.2882213.

[30] A. A. Purwita, M. Dehghani Soltani, M. Safari and H. Haas, "Impact of terminal orientation on performance in LiFi systems," *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, 2018, pp. 1-6, doi: 10.1109/WCNC.2018.8377334.

[31] B. Zhou, A. Liu and V. Lau, "Joint User Location and Orientation Estimation for Visible Light Communication Systems With Unknown Power Emission," in *IEEE Transactions on Wireless Communications*, vol. 18, no. 11, pp. 5181-5195, Nov. 2019, doi: 10.1109/TWC.2019.2934107.

[32] B. Zhou, Y. Zhuang and Y. Cao, "On the Performance Gain of Harnessing Non-Line-of-Sight Propagation for Visible Light-Based Positioning," in *IEEE Transactions on Wireless Communications*, vol. 19, no. 7, pp. 4863-4878, July 2020, doi: 10.1109/TWC.2020.2988001.

[33] M. A. Arfaoui *et al.*, "Invoking Deep Learning for Joint Estimation of Indoor LiFi User Position and Orientation," in *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 9, pp. 2890-2905, Sept. 2021, doi: 10.1109/JSAC.2021.3064637.

[34] T. Cogalan *et al.*, "5G-CLARITY: 5G-Advanced Private Networks Integrating 5GNR, WiFi, and LiFi," in *IEEE Communications Magazine*, vol. 60, no. 2, pp. 73-79, February 2022, doi: 10.1109/MCOM.001.2100615.

[35] H. Abumarshoud, L. Mohjazi, O. A. Dobre, M. Di Renzo, M. A. Imran and H. Haas, "LiFi through Reconfigurable Intelligent Surfaces: A New Frontier for 6G?," in *IEEE Vehicular Technology Magazine*, vol. 17, no. 1, pp. 37-46, March 2022, doi: 10.1109/MVT.2021.3121647.

[36] H. Abumarshoud, M. D. Soltani, M. Safari and H. Haas, "Realistic Secrecy Performance Analysis for

LiFi Systems," in *IEEE Access*, vol. 9, pp. 120675-120688, 2021, doi: 10.1109/ACCESS.2021.3108727.

[37] Y. Chen, W. Guan, J. Li and H. Song, "Indoor Real-Time 3-D Visible Light Positioning System Using Fingerprinting and Extreme Learning Machine," in *IEEE Access*, vol. 8, pp. 13875-13886, 2020, doi: 10.1109/ACCESS.2019.2961939.

[38] X. Wang and J. Shen, "Machine Learning and its Applications in Visible Light Communication Based Indoor Positioning," *2019 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS)*, 2019, pp. 274-277, doi: 10.1109/HPBDIS.2019.8735490.

[39] A. Zappone, M. Di Renzo and M. Debbah, "Wireless Networks Design in the Era of Deep Learning: Model-Based, AI-Based, or Both?," in *IEEE Transactions on Communications*, vol. 67, no. 10, pp. 7331-7376, Oct. 2019, doi: 10.1109/TCOMM.2019.2924010.

[40] Z. Qin, H. Ye, G. Y. Li and B. -H. F. Juang, "Deep Learning in Physical Layer Communications," in *IEEE Wireless Communications*, vol. 26, no. 2, pp. 93-99, April 2019, doi: 10.1109/MWC.2019.1800601.

[41] S. K. Dawood and E. K. Hamza. “Visible Light Fidelity Technology: Survey.” *Iraqi Journal of Computer, Communication, Control and System Engineering* (2021): n. pag.

[42] Abishek, R., K. Abishek, N. Hariharan, M Rakesh Vaideeswaran and C Sundara Paripooranan. “Analysis Of Machine Learning Algorithms For Wi-Fi-based Indoor Positioning System.” *2019 TEQIP III Sponsored International Conference on Microwave Integrated Circuits, Photonics and Wireless Networks (IMICPW)* (2019): 218-222.

[43] M. Leba et al. “LiFi — The path to a new way of communication.” *2017 12th Iberian Conference on Information Systems and Technologies (CISTI)* (2017): 1-6.

[44] M. A. Arfaoui *et al*., "Invoking Deep Learning for Joint Estimation of Indoor LiFi User Position and Orientation," in *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 9, pp. 2890-2905, Sept. 2021, doi: 10.1109/JSAC.2021.3064637.

[45] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang and K. Huang, "Toward an Intelligent Edge: Wireless Communication Meets Machine Learning," in *IEEE Communications Magazine*, vol. 58, no. 1, pp. 19-25, January 2020, doi: 10.1109/MCOM.001.1900103.

[46] Y. Sun, M. Peng, Y. Zhou, Y. Huang and S. Mao, "Application of Machine Learning in Wireless Networks: Key Techniques and Open Issues," in *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3072-3108, Fourthquarter 2019, doi: 10.1109/COMST.2019.2924243.

[47] H. Q. Tran and C. Ha, "High Precision Weighted Optimum K-Nearest Neighbors Algorithm for Indoor

Visible Light Positioning Applications," in *IEEE Access*, vol. 8, pp. 114597-114607, 2020, doi: 10.1109/ACCESS.2020.3003977.

[48] E. W. Lam and T. D. C. Little, "Indoor 3D Localization with Low-Cost LiFi Components," *2019 Global LIFI Congress (GLC)*, 2019, pp. 1-6, doi: 10.1109/GLC.2019.8864119.

[49] S. M. Kouhini *et al*., "LiFi Positioning for Industry 4.0," in *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 27, no. 6, pp. 1-15, Nov.-Dec. 2021, Art no. 7701215, doi: 10.1109/JSTQE.2021.3095364.

[50] N. J. Sattigiri, G. Patel, S. Mondal, R. Pal and S. Prince, "Li-Fi based Indoor Positioning System," *2019 International Conference on Communication and Signal Processing (ICCSP)*, 2019, pp. 0290-0293, doi: 10.1109/ICCSP.2019.8697974.

[51] X. Wang and J. Shen, "Machine Learning and its Applications in Visible Light Communication Based Indoor Positioning," 2019 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS), 2019, pp. 274-277, doi: 10.1109/HPBDIS.2019.8735490.