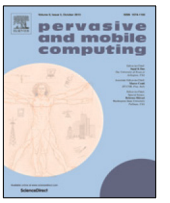
Pervasive and Mobile Computing 81 (2022) 101548

Contents lists available at ScienceDirect

Pervasive and Mobile Computing

journal homepage: www.elsevier.com/locate/pmc



Towards a better indoor positioning system: A location estimation process using artificial neural networks based on a semi-interpolated database

Batoul Sulaiman [a], Emad Natsheh [b,*], Saed Tarapiah [c]

[a] Department of Computerized Mathematics, An-Najah National University, Nablus, Palestine
[b] Department of Computer Engineering, An-Najah National University, Nablus, Palestine
[c] Department of Telecommunication Engineering, An-Najah National University, Nablus, Palestine

ARTICLE INFO

Article history:
Received 10 June 2021
Received in revised form 18 January 2022
Accepted 19 January 2022
Available online 29 January 2022

Keywords:
Fingerprinting
Received signal strength indicator
Biharmonic spline interpolation
Feed forward back propagation neural network
Generalised regression neural network

ABSTRACT

The Wi-Fi-fingerprinting positioning method is used widely in indoor positioning environments due to its simplicity and wide coverage. However, in the offline phase of the method, the collection process is a fundamental and critical step that requires time and effort. Moreover, the location estimation process, which is executed in the second Wi-Fi-fingerprinting phase (online phase), needs to be accurate enough to guarantee efficient indoor positioning. Hence, in this work, a novel indoor location-estimation process based on a semi-interpolated radio map and artificial neural network (ANN) is presented. A mobile application is built to gather the received signal strength indicator (RSSI) fingerprinting to construct a radio map, which is then expanded with the biharmonic spline interpolation (BSI) method through the estimation of more RSSI values. A feedforward back propagation (FFBP) neural network and generalised regression neural network (GRNN) were built in the online phase for the location-estimation process. They were trained using the expanded dataset by taking the reference point (X, Y) coordinates as their desired output and using two different forms of the data as their inputs. The first inputs are the RSSI values from the 17 access points (APs) – three of the APs have dual-band i.e, support both 2.4 and 5 GHz – and the second input is based on a selected set of APs, which produce a high level of acceptable RSSI and their coordinates. A comparison between these two models was done. The results show that FFBP outperforms GRNNs in terms of structure simplicity, while GRNNs achieved more accurate prediction results with an average distance error of up to 0.48 m. Hence, our proposed methodology leverages building a simple neural network topology that has good location estimation results for indoor positioning in a low-cost localisation process.

1. Introduction

In several application domains, users' location knowledge became a critical issue. Users' locations are estimated using localisation systems categorised as indoor and outdoor systems. Various solutions, such as global positioning systems (GPSs), are adopted in outdoor environments; these are considered unreliable in indoor environments due to the unavailability of lines of sight and the existence of obstacles such as walls. As a result, several research studies have been

* Corresponding author.
E-mail address: e.natsheh@najah.edu (E. Natsheh).

https://doi.org/10.1016/j.pmcj.2022.101548
1574-1192/

conducted on location estimation in indoor environments to develop efficient and accurate indoor positioning systems (IPSs).

IPSs can be classified based on the information acquired from the wireless signals with range-based and range-free techniques. Range-free techniques are cost-effective because there is no need for additional hardware for the location-estimation process, as the location is estimated based on distance-approximation algorithms [1]; among range-free localisation algorithms, DV-Hop and centroid algorithms are considered important and popular. However, in range-based techniques, the target location is determined using trilateration or triangulation approaches based on different geometric methods to measure the distance or angle between the transmitting and receiving nodes with a high level of accuracy and robustness [1]. In fact, various methods are used to determine geometric information for range-based techniques, such as received signal strength indicator (RSSI)-based and time-based methods, which are time difference of arrival (TDoA) and time of arrival (ToA) [2]. In time-based methods, the location estimation of TDoA measurements is estimated using the cross-correlation technique, and the ToA-based method measures the propagation time from the transmitter to the receiver [2]. Generally, the emitted signals have poor penetration capabilities; as a result, to overcome this problem, a cooperative localisation system that enables the exchange of the information ranged between the neighbouring nodes can be used [3].

Over time, various range-based localisation techniques have been developed. For instance, GPS is one of the most popular range-based localisation techniques using ToA and TDoA measurements but is considered an energy-consuming technique and inefficient in indoor environments. Another is the global system for mobile communications (GSM), which uses RSSI and angle of arrival (AoA) methods. Besides, the ultra-wideband (UWB) technique, which uses ToA measurements and is able to provide highly accurate range measurements [4].

For accurate range measurement in time-based methods, synchronisation is an important and fundamental requirement; therefore, cooperative network synchronisation based on the UWB technique is currently one of the most attractive topics, and rich literature is available on this topic. In [5], the authors introduce an asymptotic analysis for cooperative network synchronisation and derive performance limits for both absolute and relative synchronisation problems. Some of time synchronisation protocols (TSPs) in wireless sensor networks are discussed in [6,7], such as reference broadcast synchronisation (RBS), the time-synchronisation protocol for sensor networks (TPSNs) and the flooding time synchronisation protocol (FTSP). Moreover, localisation systems can have passive or active localisation; in [8], the author focuses on the passive type based on ToA measurements constructed over a unified-factor graph-based framework, and the Cram'er–Rao bound was derived to characterise localisation performance.

Owing to the wide range of devices equipped with a Wi-Fi adapter nowadays, Wi-Fi-based indoor localisation systems are adopted based on RSSIs, so RSSI-based indoor localisation systems are more cost-effective and have less complexity than time-based methods due to less demand for additional hardware [9]. Existing Wi-Fi-based localisation methods can be put into two categories: Wi-Fi-fingerprinting and Wi-Fi-ranging [10]. The latter estimates the user's location directly using distance to the access point (AP), which is inefficient and impractical in indoor environments like buildings because of the influence of people, wall reflections and signal occlusions [11]. In Wi-Fi-fingerprinting methods, it is fundamental to associate a fingerprint to a specific position to be able to use it later to identify the position. The Wi-Fi-fingerprinting method consists of two phases: offline and online. The data-collection process is executed in the offline phase to build a fingerprint database. The location-estimation process is executed in the online phase using various localisation algorithms. Wi-Fi fingerprinting methods can be categorised into database-based [12] and neural network-based methods [12].

- Category #1 — Database-based approach: This type relies on the collection of RSSI values from different APs in several locations named by reference points (RPs) and the saving of these in a database. Then, a target mobile collects RSSI values at a random location and compares them to stored ones in the pre-recorded database to estimate the coordinates of the user's location.
- Category #2 — Neural network-based approach: The RSSI fingerprinting method, combined with machine-learning algorithms, is a promising indoor localisation solution. Therefore, solutions based on machine-learning algorithms are highly recommended in various localisation systems. In this approach, signal strength is recorded at each RP and stored in a database called a radio map. Later, this training samples from this database are used for neural-network models. Artificial neural networks (ANNs) are characterised by the capability to learn by themselves and approximate highly nonlinear models to produce the desired outputs. Recently, various ANN-localisation solutions, such as multilayer perceptrons (MLPs) [13], convolutional neural networks (CNNs) [14], recurrent neural networks (RNNs) [15] and generalised regression neural networks [16], have been proposed in the indoor localisation field.

This study adopts the neural-network-based approach and aims to find an ANN model that is able to estimate accurate locations inside a university college; the availability of this positioning information means that students, staff and visitors will never get lost, as they only need to glance at their mobile phones to see their location, and it gives them the ability to reach their destination easily. The RSSI data-collection process from various RPs is critical in the localisation scenarios for an ANN model to be an efficient IPS. Therefore, this paper introduces an RSSI-based IPS, located on the second floor of the engineering college at An-Najah National University [17], to generate a radio map consisting of real RSSI measurements, using the biharmonic spline interpolation (BSI) method, to expand the pre-recorded radio map. Then, two different ANNs, which are the feedforward neural network (FFNN) and generalised regression neural network (GRNN) models, are used to estimate user location accurately with two different inputs. The results show that the proposed ANN models have low

complexity and improved localisation accuracy. The positioning system presented in this paper can be applied in other environments such as airports, indoor parking, hospitals and shopping malls.

This paper is organised as follows: Section 2 introduces the related works. Section 3 provides a detailed description of the proposed system. Section 4 illustrates the methodology of the experiment. The results, discussion and comparison between this paper and some of the state-of-the-art studies are described in Section 5. Finally, in Section 6, a conclusion is given.

2. Related work

The main challenges in fingerprinting systems are the effort and time required to execute the data-collection process in the offline phase in order to construct the radio map and the time-consuming procedure of searching through the stored fingerprint samples in the database during the online fingerprinting phase to estimate user location. Recent literature introduces different methods of mitigating the required effort and time of the offline phase. In [18], manual calibration is avoided through the use of a collaborative approach that uses the collected RSSI values at fixed nodes to achieve 2.5 m of indoor localisation accuracy. In [19], the author proposes an algorithm based on an improved double-peak Gaussian distribution to generate Wi-Fi fingerprints. A new RSSI prediction model with a fingerprint calibration procedure is used in [20]. In [21], the kriging interpolation method was used for database generation, which gives good estimation results. The author of [22] proposes a crowdsourcing method to construct a manual radio map and uses the inverse distance weighting (IDW) interpolation method to generate an interpolated radio map, and in [23], the simultaneous localisation and mapping (Wi-Fi SLAM) method is used although it suffers from a heavy computational load, it has an accurate result. In [24], the author proposes a Htrack map-matching system based on a hidden Markov model (HMM) for the location estimation process in two different indoor environments; in general, the map-matching method is computationally expensive. However, there are such methods that can run on a mobile device such as in [25], a mobile application based on Wi-Fi fingerprinting is proposed for user location estimation in an indoor environment, in addition, the author in [26] introduces a Cost-effective Indoor Localisation that combines the benefits of two localisation techniques, WiFi and Kinect, into a single algorithm using low-cost sensors. In [27], the author considers the operationally deployed Live Labs server-side Wi-Fi localisation system which uses the classical RADAR algorithm tackle some limitations such as low cardinality and outlier elimination and achieved 80+% reduction in the overall occupancy estimation error. In addition, two methods of localisation are presented in [28–31] that are static and dynamic methods. For the online fingerprinting phase, traditional localisation algorithms are used in several studies; the author of [32] introduces a comprehensive study of different traditional algorithms. The use of machine-learning algorithms for indoor localisation has been on the rise due to their efficient and accurate estimation results. Therefore, for the location-estimation process in the online phase, different kinds of ANN algorithms have been applied, such as FFNNs [33,34], CNNs [35], radial basis neural networks (RBNNs) [36] and deep neural networks (DNNs) [37]. The author of [38] uses CNN-based Wi-Fi fingerprinting for location estimation in multi-storey buildings and compares it with DNNs — this resulted in higher accuracy with less data. In [39], MLPs and GRNNs are used to estimate user location through ToA measurements. Different machine-learning approaches are used in [40]: ANNs and support-vector regression are used to locate APs rather than the mobile user. Table 1 shows parts of recent works that use ANN in localisation. There are also machine-learning algorithms that have been used for indoor fingerprinting, such as the gauss process model [41] and nearest-neighbours [42], random-forest [43] and GD-based methods [44].

In the previously mentioned works, there is not a method that considers both the urgent need to reduce the time and effort of the offline phase and the use of an efficient machine-learning method with high location-estimation accuracy. Also, the integration of real RSSI measurements, BSI-method measurements and FFNN or GRNN models was not adapted for accurate localisation in Wi-Fi environments. A comparison between this paper and some of the related work is discussed in Section 5. The main contributions of this paper can be summarised as follows:

- The proposed system was verified with real location coordinates and RSSI data, which were collected from An-Najah National University using a proposed mobile application, which has the ability to collect accurate RSSI fingerprint values and corresponding records (RSSI fingerprint, name of AP, MAC address (MA) of AP, etc.)
- A comprehensive approach that considers the optimisation of the offline and online fingerprinting phases was proposed.
    - In the offline phase: The Wi-Fi-fingerprinting method was used to build a dataset, and BSI surfaces were employed to estimate more fingerprint values and generate a denser database.
    - In the online phase: Two neural-network algorithms, namely the feedforward back propagation (FFBP) neural network and GRNN, were used for the location-estimation process. They were trained using a semi-simulated dataset with two different input patterns.
- The deployed AP location and the valid received signal strength were taken into consideration during the neural-network training process. In addition, the deployed single-band and dual-band APs supported both 2.4 and 5 GHz.
- We evaluated the positioning errors using different performance criteria, and accurate estimated locations with average distance errors of up to 0.48 m were obtained.

Table 1
Shows part of the recent existing works that use ANN to improve localisation.

References	ANN Type	Hidden Layer	Input	Output
[45]	ANFIS	3	RSS, ZigBee	Distance
[46]	CNN	8	RSS map	Room index and location
[47]	CNN	3	RSS map	Location
[48]	CNN	8	AoA	Location
[49]	CNN	3	CIR	Location
[50]	CNN	2	RSS, photodiode	Cell index
[51]	CNN	10	Sound	Region index
[52]	CPN	2	RSS, WiFi	Region index
[53]	FFNN	1–3	RSS, WiFi	Location
[54]	FFNN	2	RSS, RFID	Location
[55]	GAN	3	RSS, WiFi	Location
[56]	RBNN	1	RSS, BLE	Location
[57]	RNN	10	CSI, WiFi	NLoS identification
[58]	SCG and RBP	2–4	RSS, WiFi	Room index and location
[59]	TDNN	3	Sound	AoA

ANN types: ANFIS-adaptive neural fuzzy inference system; CNN-convolution neural network; CPN-counter-propagation neural network; FFNN-feed-forward neural network; GAN-generative adversarial neural network; RBNN-radial basis neural network; RNN-recurrent neural network; SCG-scaled conjugate gradient; RBP-resilient back propagation; TDNN-time delay neural network.

3. Proposed system

This section introduces the indoor location-estimation process, which was used in four main phases:

- Phase #1: Online and offline Wi-Fi-fingerprinting
- Phase #2: Semi-interpolated database construction
- Phase #3: Neural-network selection
- Phase #4: Model evaluation

Fig. 1 shows the proposed system used for the indoor location-estimation process.

3.1. Location fingerprinting

The Wi-Fi fingerprinting localisation system is divided into two phases [12]. In the first phase (offline 'training phase'), the coordinates of the training points (reference points), with their corresponding RSSI values from the deployed APs, are recorded and stored in a central database. In the second phase (online 'estimation phase'), a specific algorithm is used to estimate user location by matching the observed RSSI at its location with the RSSI values from the pre-recorded database in the offline phase.

■ **Fingerprint collection**

Fingerprinting is a highly accurate localisation method that has low complexity and high applicability in many complex indoor environments [60]. Fingerprint collection includes the scanning of APs within the vicinity based on the beacon frames, and fingerprints are periodically sent through APs for synchronisation. This method involves collecting the RSSIs for APs detected during the scanning phase. In this study, the fingerprint collection tool is a proposed mobile application used to collect multiple RSSI samples at each RP from the deployed APs. Basically, by determining the RP name, name of the section containing this RP and determining the number of RSSI samples to be collected at each RP from the deployed AP, the database is stored as a comma-separated values (CSV) file on the server. A screenshot of the proposed mobile application used to generate the database is shown in Fig. 2.

The recorded database consists of the RSSI values from the deployed APs as fingerprints for the RP. Other related information saved in the database is the name of the region containing the RP, the name and the MA of the AP, which transmits beacon frames periodically, the collection date and time for every RSSI sample and its number and the (X, Y) coordinates of the RPs.

At each RP, the mobile application allows the user to choose the number of RSSI samples in the deployed scenarios; 60 samples are selected, and the total number of RSSI records is 67000. However, only the average values of the multiple RSSI samples are stored in the database [61]. Eq. (1) shows how to find the mean of n RSSI values from the ithAP to the jth RP.

$$\overline{RSSI} = \frac{1}{n}\sum_{t=1}^{n} RSSI_{ij}^{t} \tag{1}$$

Where, $RSSI_{ij}^{t}$ is the tth element of the collected $RSSI_{ij}$ samples.

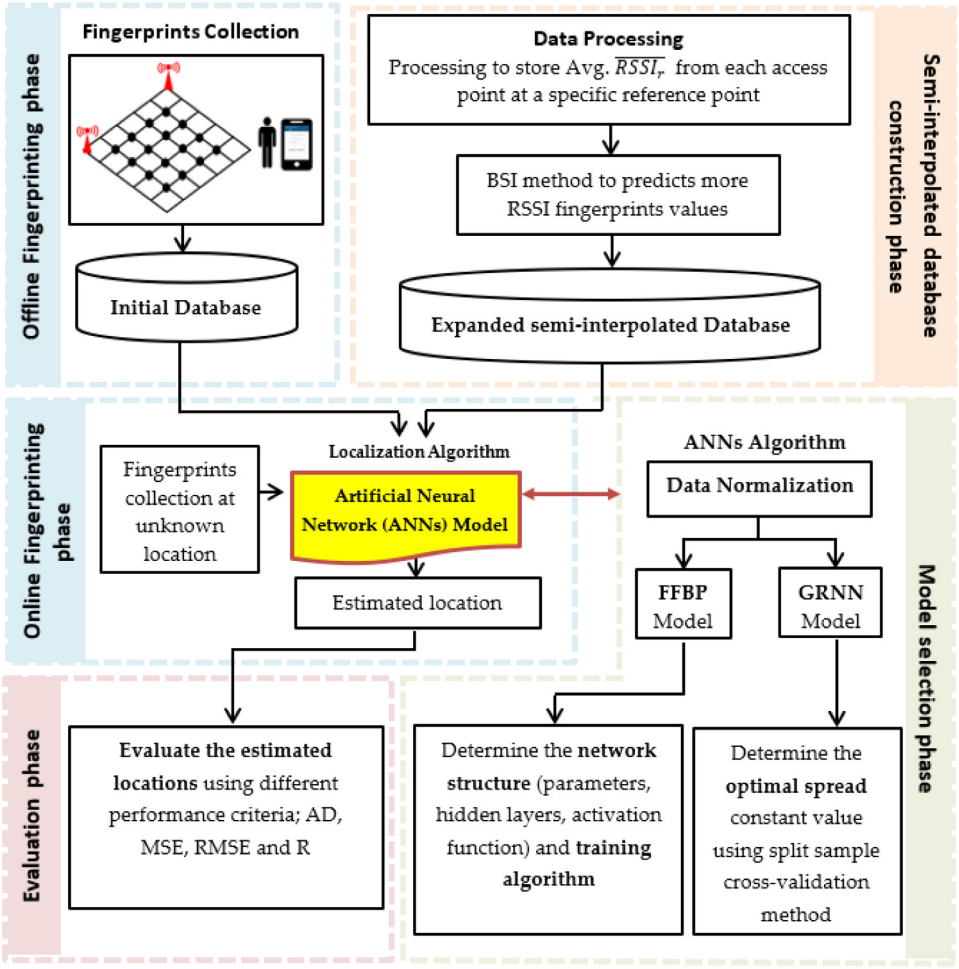
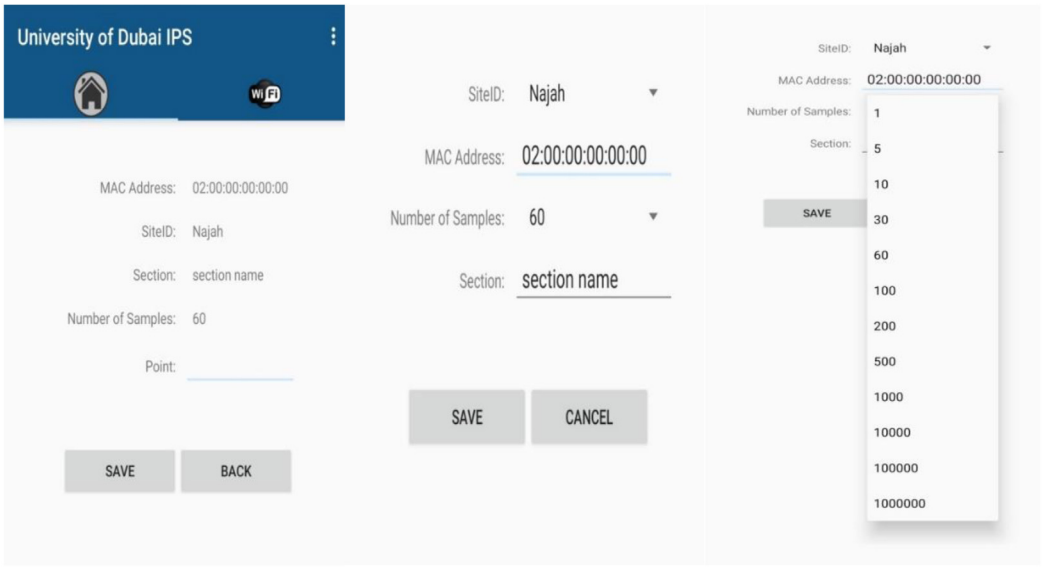


Fig. 1. Diagram of the proposed indoor location estimation system.



(a) (b) (c)

Fig. 2. Mobile application interface (a) main page (b) section input page (c) choosing number of sample page.

3.2. The biharmonic spline interpolation method for database expansion

In this study, during the fingerprinting online phase, a neural-network model will be used as a localisation method to estimate the unknown target location, depending on the generated database. Neural-network estimation accuracy is affected by the size of the radio map — a larger radio map results in a more accurate location-estimation process. Moreover, a lot of effort and time is needed to build a database in the fingerprinting offline phase, which is considered a drawback in the localisation process.

A suggested way to estimate more data using the data already available in the offline phase is the use of numerical methods. Different numerical methods have been introduced by researchers [62–64]; one of these is the BSI method [65]. This method is used to find the smoothest surface passes exactly through the input points. The interpolated surface is described by Sandwell and David as a linear combination of Green's functions. The solutions of Green's function for m-dimension biharmonic operator imply that the surface f(u) can be expressed as in Eq. (2) [65]:

$$f(u) = \sum_{i=1}^{n} \alpha_i G(u, u_i) \tag{2}$$

where, G is the Green's function, u is the output location, u_i is the location of the *ith* data constraint, and α_i are the associated weights. A detailed description of the mathematical aspects of the BSI method can be found in [65].

An advantage of this method is that there are no rounding errors at the checkpoints. Moreover, the increasing number of used checkpoints mitigates the interpolation error value, which is not true of all other interpolation methods, such as polynomial interpolation. Therefore, in the fingerprint offline phase, BSI is a suitable method of constructing a denser database of RSSI fingerprints, as illustrated in [66]; in fact, the author proves that there is a small difference between the simulated and actual values

3.3. Artificial neural network

An ANN is an information processing model that has learning ability and high prediction accuracy in different fields. ANN models consist of interconnected processing elements called neurons arranged in different layers. A training process is used with the ANN model to find the relationships between the neural network inputs and targets. Among the existing ANN models, two efficient types are used in this paper: the FFBP neural network and GRNN.

■ **Feedforward back propagation neural network**

The FFBP neural network is an efficient and popular neural-network model used in many engineering applications. FFBP networks consist of neurons arranged in three layers: input, output and hidden layers between them. The main responsibility of the hidden layers is to execute intermediate computations such that the hidden neuron receives the input value directly from the input layer and associates it with a specific random weight, uniformly distributed inside range $\left(\frac{-2.4}{f_i}, \frac{+2.4}{f_i}\right)$, where f_i is the total number of inputs of neuron i in the network. With weighted linear summation (X), as shown in Eq. (2), each neuron in the hidden layer transforms the previous layer's values, which are then evaluated using an activation function.

$$X = \sum x_i w_i + b_i \tag{3}$$

Where, x_i is the input values, w_i is the corresponding weights and b_i is the bias.

Each network's layer has its own specific activation function; these limit the layer's output values in a specific interval. A suitable choice of activation function improves the neural network's results. Examples of these are the sigmoid, hyperbolic tangent (Tanh), rectified linear unit (ReLU) and linear function [67].

For efficient training, backpropagation must be based on training algorithms; their role is to iteratively adjust the weights and bias throughout the training process in order for the network to improve its performance. Training algorithms, such as gradient descent (GD), are also called steepest descent [68]; GD is a simple iterative algorithm that requires many iterations, which leads to a slow training process. Conjugate gradient (CG) [69] is a line search algorithm, along with conjugate direction, and has faster convergence than GD. Scaled conjugate gradient (SCG), developed by Moller [70], avoids consuming time since a line search is not performed at each iteration as with the CG algorithm. Bayesian regularisation (BR) [71] is considered a robust mathematical process and eliminates the need for a validation set. However, the FFBP model is difficult to overfit when using this type of training algorithm. The Levenberg–Marquardt (LM) [72] is one of the most robust and fastest training algorithms since it shortens the iteration process; it also balances training speed and stability.

The FFBP model needs to determine the number of neurons and hidden layers, type of training algorithm and activation function. This depends on the relationship's complexity between the inputs and target data. Section 3.2 illustrates the pathways for the model selection process.

■ **Generalised regression neural network**

GRNN is a probabilistic ANN model related to the RBNN that was introduced in 1991 by Specht [73]. GRNNs needs a larger number of neurons than FFBP neural networks but require a fraction of the designing time to be constructed. In

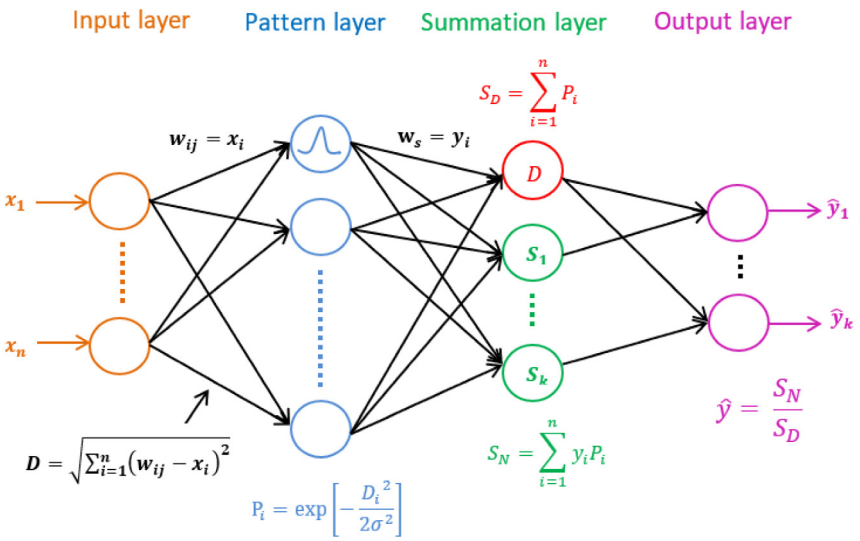


Fig. 3. GRNN structure.

the GRNN optimisation process, only one parameter needs to be determined, called the spread constant (σ); no iterative procedure is needed, and it is a fast and stable method [74]. During model construction, spread-constant selection is a critical issue: If an appropriate value is not chosen for σ, the required training results will not be achieved. Fig. 3 illustrates the architecture of the GRNN, which contains four components:

1. Input layer: contains the original data to be submitted to the next layer.
2. Hidden layer (pattern layer): The number of hidden neurons is equal to the number (n) of learning samples, such that each hidden neuron corresponds to one learning sample. A radial basis function (gaussian transformation function) is applied to the values received from the input layer, given by Eqs. (3) and (4). The weights (w_{ij}) between the input and pattern layers are the values of the input parameters. The pattern layer is fully connected to the third layer, and its output is the distance between the input and the stored patterns.

$$\mathrm{P}_i = \exp\left[-\frac{\mathrm{D}_i^2}{2\sigma^2}\right], i = 1, 2, \ldots, n \tag{4}$$

$$\mathrm{D}_i^2 = (x - x_i)^T (x - x_i) \tag{5}$$

Where, P_i is the *ith* hidden neuron's output, σ the spread constant, x the network input vector and x_i the corresponding learning sample for the *ith* neuron. D_i is the Euclidean distance between the input vector and training (learning) samples.

3. Summation layer: Each hidden neuron is fully linked to this layer (summation layer). It has two different types of summation:

   - The S-summation neuron, which determines the sum of the weighted outputs of the pattern layer.
   - The D-summation neuron, which determines the unweighted outputs of the pattern neurons.

   The value of target output y_i, which corresponds to the *ith* input value, is considered the connection weight (w_s) between the *ith* hidden neuron and the S-summation neuron.
4. Output layer: The predicted result is derived from this layer by dividing the S-summation neuron's (S_N) output by the D-summation neuron's (S_D) output as in Eqs. (5) and (6):

$$S_N = \sum_{i=1}^{n} y_i P_i \tag{6}$$

$$S_D = \sum_{i=1}^{n} P_i \tag{7}$$

3.4. Evaluation criteria

Different performance criteria [74–76] were applied to evaluate the estimated locations using the two ANN models as given in the following equations:

(a) The average distance error (AD): Distance error is a measure of how far apart the points are, and the AD is calculated by averaging the distance errors between all the points, as in Eq. (7).

$$\mathrm{AD} = \sum_{i=1}^{n} \frac{1}{n} \sqrt{(x-\hat{x})^2 + (y-\hat{y})^2} \tag{8}$$

Where x and y is the actual coordinate, $\hat{x}$ and $\hat{y}$ is the estimated coordinates and n is the total number of data.

(b) Mean square error (MSE): Measures the average of the squares of the errors; its value indicates the difference between the actual and predicted value. MSE is expressed in Eq. (8).

$$\mathrm{MSE} = \frac{1}{n} \sum_{i=1}^{n} (p-\hat{p})^2 \tag{9}$$

Where p is the actual value and $\hat{p}$ is the predicted value and n is the total number of data.

(c) Root mean square error (RMSE): A quadratic score that measures the average magnitude of the error, as expressed in Eq. (9).

$$\mathrm{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (p-\hat{p})^2} \tag{10}$$

Where p is the actual value and $\hat{p}$ is the predicted value and n is the total number of data.

(d) Mean absolute error (MAE): The average magnitude of the differences between the actual and predicted values, as in Eq. (10).

$$\mathrm{MAE} = \frac{1}{n} \sum_{i=1}^{n} |p-\hat{p}| \tag{11}$$

Where p is the actual value and $\hat{p}$ is the predicted value and n is the total number of data.

(e) Correlation coefficient (R): A statistical measure that indicates the strength of a relationship between two variables; a high R indicates a strong while a small R means a weak relationship. R is expressed in Eq. (11).

$$\mathrm{R} = \frac{n \sum_{i=1}^{n} p_i \hat{p}_i - \sum_{i=1}^{n} p_i \sum_{i=1}^{n} \hat{p}_i}{\sqrt{(n \sum_{i=1}^{n} p_i^2 - (\sum_{i=1}^{n} p_i)^2)(n \sum_{i=1}^{n} \hat{p}_i^2 - (\sum_{i=1}^{n} \hat{p}_i)^2)}} \tag{12}$$

Where p is the actual value and $\hat{p}$ is the predicted value and n is the total number of data.

The distance measurements (AD, MSE, RMSE and MAE) help determine the accuracy of the prediction values, compared to the actual values, and small values mean the prediction model is close to the actual values.

4. Experiment analysis

4.1. Database generation

This subsection illustrates the indoor environment, which is on the secondfloor of the engineering college at An-Najah National University, which covers an area of approximately 37 × 32 m^2. Fig. 4 (a and b) shows a map (2D and 3D) of the study area. There are 17 APs (three are dual-band) distributed in the area, as illustrated in Fig. 4 with green symbols. The type of the deployed APs is TP-link and all the APs are installed with an approximate height of 2–3 m. For the training phase, Samsung galaxy A70, android version 10 mobile phone is installed on a stand, is used as a receiver in the offline fingerprinting phase and moves along a route of RPs 1.35 m apart, as illustrated in Fig. 4(c), to collect the RSSI readings using the installed mobile application to construct the database. Processing steps to find the average of the collected RSSI values at each RP from each AP are done, and the initial database is constructed. In the estimation phase, RSSI values correspond to 20 random testing points are gathered between the fixed RPs in the same manner as the data gathering conducted in the training phase, 60 RSSI samples are gathered for each testing point from each installed APs, the sample size of the testing points is used as an out-of-sample data to check the accuracy of the proposed ANN models.

The resulting initial database obtained from the experiment is expanded using the BSI method to generate a denser database and attain effective location estimation results using an ANN. The experimental RSSI values, which correspond

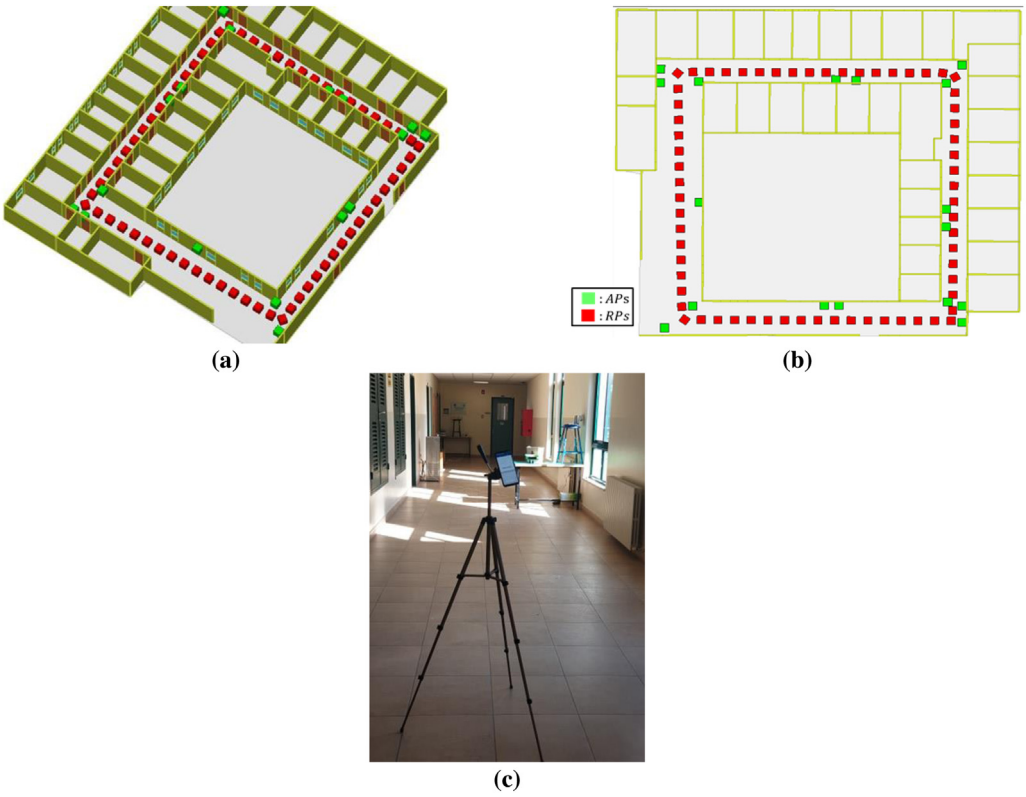


Fig. 4. (a) 3D map of the study area (b) 2D map of the study area. The APs and RPs locations is showed in (a) and (b). (c) The used mobile phone in the study area.

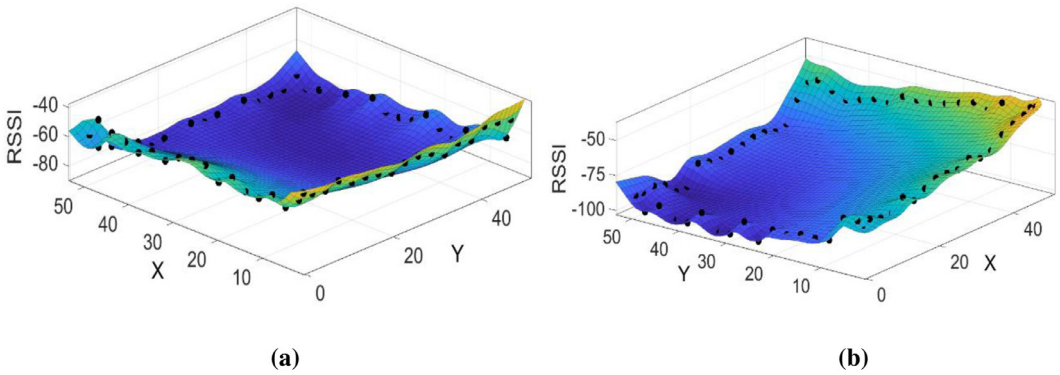


Fig. 5. (a) interpolated surface generated by BSI method for AP(1) (b) interpolated surface generated by BSI method for AP(2).

to 64 RPs from each AP, are used to generate the interpolated surfaces for each APs such that each surface meets one APs and covers the region under study. Fig. 5 shows the generated surfaces for two of the deployed APs.

In Fig. 5, the black dots in the surfaces represent RPs that have actual RSSI values. Using these generated surfaces, the values of the RSSI measurements at other points of the surface are gathered. As a result, an expanded semi-interpolated database is constructed; it consists of 190 RPs, spaced 0.45 m apart such that 64 RPs have actual RSSI measurements, and the remaining 126 have interpolated RSSI values obtained using the BSI surfaces. The database-generation flow is shown in Fig. 6.

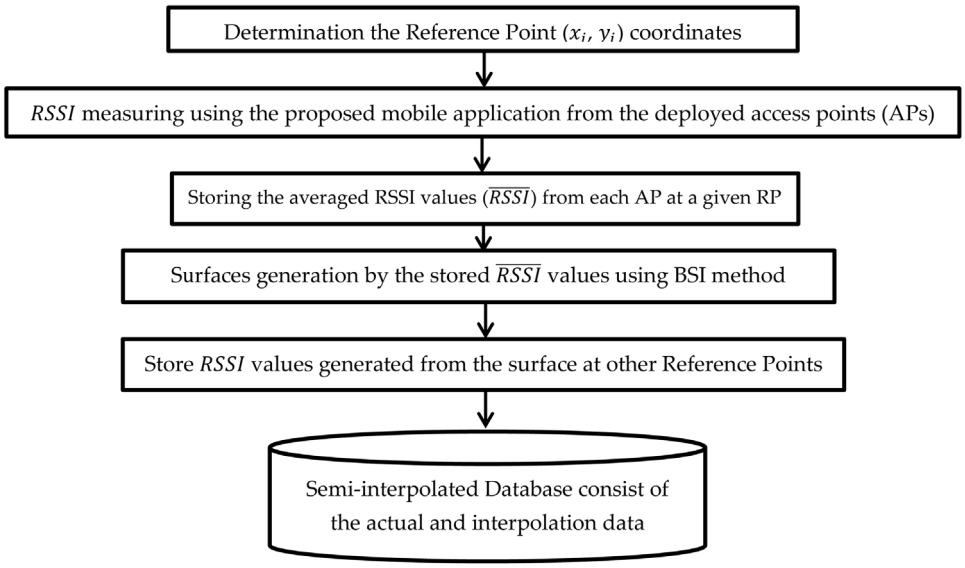


Fig. 6. Database generation process.

4.2. Model selection and evaluation

■ **Feedforward back propagation neural network**

To determine the best neural network configuration, several trials should be made by changing the number of hidden layers, number of neurons, activation function and training algorithm. Each trial should be repeated several times to obtain the best training result. Furthermore, two different FFBP neural network structures are presented.

(1) Input pattern based on all the deployed APs

The first proposed structure is based on the use of all the RSSI values from all the deployed 17 APs (three are dual-band), such that each FFBP input neuron corresponds to one AP. Hence, this structure has 20 neurons in the input and two in the output layer, which represent the (X,Y) coordinates. The input is RSSI $[\mathrm{RSSI}_{AP1}, \mathrm{RSSI}_{AP2}, \ldots, \mathrm{RSSI}_{AP20}]$ and the output is the location coordinates $[\mathrm{XY}]^{\mathrm{T}}$. Tables 2 and 3 show the executed trials used to select the best FFBP model for in-sample and out-of-sample data, respectively.

The determination of the optimal number of hidden neurons is a critical decision in the network-constructing phase. A neural network with few hidden neuron numbers does not have the ability to model complicated data, so the generated structure has inefficient prediction results. Several hidden neurons provide the network with a good training process and good prediction results for the trained data (in-sample), but it loses the ability to give correct predictions for new data that has not been trained previously (out-of-sample); its ability to generalise is badly affected. Therefore, a compromise must be reached [77]. From Tables 2 and 3, the optimal number of hidden neurons was determined with network-performance trials.

As shown in Tables 2 and 3, the best structure was obtained in trial 5, with one hidden layer with 20 neurons, Tansig as the hidden activation function and LM as the training algorithm. As illustrated, in the in-sample phase, the MSE for training, validation and testing this structure is 1.17×10^{-22}, 0.101 and 0.137, respectively. In the out-of-sample phase, the AD, MSE, RMSE, MAE and R is 0.96, 2.52, 1.58, 1.18 and 0.996 m, respectively.

(2) Input pattern based on a selective set of APs, which produce a high level of acceptable RSSI and their coordinates.

The second proposed structure is based on a selective set of APs, which produce high-level, acceptable RSSIs of the corresponding RPs. In such a structure, six APs generate more than −75 dbm in the reference-related points. The selective six APs and their coordinates are taken as input into the structure, which leads to 18 input and two output neurons, which represent the location coordinates. The input is $[RSSI_{AP1}, X_{AP1}, Y_{AP1}, RSSI_{AP2}, X_{AP2}, Y_{AP2}, \ldots, RSSI_{AP6}, X_{AP6}, Y_{AP6}]^T$, and the output is the $[XY]^T$. Tables 4 and 5 show the executed trials used to select the best FFBP model for in-sample and out-of-sample data, respectively.

In Table 4, the number of hidden layers and neurons is varied and the best one can be determined with the best network performance, which has the minimum MSE value for the training, validation, and testing sets. Trials 4 and 18 have the minimum MSE values for the three sets. The parameters used in these trials are: Tansig as the hidden activation function and LM as the training algorithm. However, in trial 4, the structure has a single hidden layer with 15 neurons, but the network structure in trial 18 has two hidden layers with 25 neurons at each layer. In addition, in the results for

Table 2
Trials to select the best FFBP (20 input) neural network structure.

FFBP with 20 inputs neurons									
Trial	Hidden Neurons	Activation Function	Training Algorithm	MSE Training	MSE Validation	MSE Testing	R	Epoch	Train, Valid, Test
1	5	Logsig	LM	0.1584	1.3372	0.361	0.9994	9	60%, 20%, 20%
2	10	Tansig	LM	0.4424	1.7067	1.0138	0.9991	10	60%, 20%, 20%
3	15	Tansig	LM	0.029	0.5746	0.5403	0.99968	8	60%, 20%, 20%
4	15	Logsig	LM	0.0774	0.2767	0.5729	0.99974	12	60%, 20%, 20%
5	**20**	**Tansig**	**LM**	**1.1779×10^{-22}**	**0.1077**	**0.1373**	**0.99993**	**12**	**60%,20%,20%**
6	20	Tansig	BR	1.885×10^{-15}	NaN	0.0108	1	1849	70%, 30%
7	25	Logsig	SCG	3.0234	5.105	8.0858	0.9945	38	70%, 15%, 15%
8	25	ReLU	BR	0.0091	NaN	0.2507	0.99993	69	80%, 20%
9	30	Tansig	BR	1.2825×10^{-19}	NaN	0.0297	0.99999	1511	70%, 30%
10	30	Logsig	SCG	2.2122	3.6466	5.6793	0.9958	87	60%, 20%, 20%
11	35	Tansig	LM	1.0603×10^{-9}	0.0815	0.0489	1	6	80%, 10%, 10%
12	40	Tansig	LM	0.0073	0.4325	0.4929	0.99975	4	70%, 15%, 15%
13	[10, 10]	[ReLU, ReLU]	LM	0.0122	0.1871	0.4302	0.99983	14	60%, 20%, 20%
14	[5, 5]	[tansig, tansig]	LM	0.003	0.2403	0.4485	0.99982	32	60%, 20%, 20%
15	[10, 5]	[ReLU, tansig]	BR	0.033	NaN	0.1323	0.99992	90	70%, 30%
16	[15 15]	[tansig, tansig]	LM	0.0039	0.337	0.2671	0.99984	6	60%, 20%, 20%
17	[25 25]	[tansig, tansig]	LM	1.132×10^{-14}	NaN	0.0253	0.99999	915	70%, 30%
18	[15 22]	[logsig, logsig]	SCG	0.2258	0.3628	0.7974	0.99956	232	70%, 15%, 15%
19	[10 10 10]	[tansig, tansig, tansig]	SCG	0.1918	0.4822	0.4355	0.99964	124	70%, 15%, 15%
20	[20 20 10]	[tansig, tansig, tansig]	LM	0.002	0.4672	0.1484	0.99984	6	60%, 20%, 20%

Activation Functions: tansig-Hyperbolic tangent sigmoid transfer function; logsig-Log sigmoid transfer function; ReLU-rectified linear unit activation.
Training Algorithms: LM-Levenberg Marquardt algorithm; BR-Bayesian regularisation; SCG-Scaled conjugate gradient.

Table 3
Out-of-sample testing points from FFBP (20input) neural network structures.

Out of sample testing points from FFBP (20input)					
Trial	AD (m)	MSE (m)	RMSE (m)	MAE (m)	R
1	1.3477	6.1601	2.482	1.632	0.99166
2	1.2612	4.815	2.1943	1.5179	0.9928
3	1.6078	7.7952	2.792	2.0543	0.99113
4	1.8934	11.85	3.4424	2.2718	0.98342
5	**0.96662**	**2.5256**	**1.5892**	**1.1836**	**0.99668**
6	1.1576	3.8537	1.9631	1.4929	0.99447
7	1.4252	5.6136	2.3693	1.7791	0.9916
8	1.3666	5.1466	2.2686	1.7878	0.99383
9	1.1701	4.2252	2.0555	1.4856	0.99369
10	1.256	4.3987	2.0973	1.582	0.9935
11	1.7062	8.2802	2.8775	2.2785	0.98834
12	1.6111	7.7739	2.7882	2.0841	0.989
13	1.4474	6.308	2.5116	1.5474	0.99074
14	2.1749	15.149	3.8921	2.2474	0.97633
15	1.6685	8.878	2.9796	1.7239	0.98774
16	1.2398	4.1956	2.0483	1.5224	0.99399
17	0.98608	3.1212	1.7667	1.1684	0.99532
18	0.96008	2.7576	1.6606	1.1521	0.99601
19	0.87186	2.29	1.5133	1.001	0.99695
20	1.0952	3.655	1.9118	1.2885	0.9945

the out-of-sample data from Table 5, trial 4 has better performance criteria. As a result, the best structure was obtained in trial 4 due to its simplicity and superior generalisation.

■ Generalised regression neural network

Two different GRNN models were tried based on the number of APs used as inputs in the neural network.

(1) Input pattern based on the deployed APs

This neural network structure depends on the RSSI values obtained from the 17 APs. Hence, as with the FFBP, the input is *RSSI* $[RSSI_{AP1}, RSSI_{AP2}, \ldots, RSSI_{AP20}]$ and the output is $[XY]^T$.

The spread constant affects the network's degree of generalisation. Therefore, it is important to determine its optimal value in order to generate an efficient network. Using the split-sample cross-validation method, the optimal spread-constant value was selected such that the database was split into two sets: 75% of the data used a training set, the other

Table 4
Trials to select the best FFBP (18 input) neural network structure.

FFBP with 18 inputs neurons									
Trial	Hidden Neurons	Activation Function	Training Algorithm	MSE Training	MSE Validation	MSE Testing	R	Epoch	Train, Valid, Test
1	5	Logsig	LM	2.5408	2.7222	3.949	0.9962	7	60%, 20%, 20%
2	10	Tansig	LM	0.3289	0.6578	1.5588	0.9991	41	60%, 20%, 20%
3	10	ReLU	LM	0.7106	1.4479	2.6811	0.9983	19	60%, 20%, 20%
4	**15**	**Tansig**	**LM**	**0.1645**	**0.8624**	**0.701**	**0.9995**	**43**	**70%,15%,15%**
5	15	Logsig	LM	0.1476	1.5994	0.9151	0.9992	11	60%, 20%, 20%
6	15	Tansig	SCG	1.78	3.19	2.01	0.9973	92	70%, 15%, 15%
7	20	Logsig	SCG	2.8813	4.19	4.6848	0.9953	66	60%, 20%, 20%
8	25	Tansig	LM	0.0117	1.9246	2.0996	0.9989	11	60%, 20%, 20%
9	25	ReLU	BR	1.5184	NAN	3.8957	0.9972	85	70%, 30%
10	25	Tansig	BR	2.56×10^{-7}	NAN	1.6069	0.9994	1000	70%, 30%
11	30	Tansig	LM	5.65×10^{-2}	2.6799	2.6667	0.9986	17	60%, 20%, 20%
12	[10, 10]	[ReLU, ReLU]	LM	0.8509	2.8353	4.9989	0.9972	16	60%, 20%, 20%
13	[10, 10]	[tansig, tansig]	BR	6.63×10^{-6}	NAN	1.6293	0.9995	1000	60%, 20%, 20%
14	[5, 5]	[tansig, tansig]	LM	1.3484	3.9539	2.8053	0.9971	23	60%, 20%, 20%
15	[10, 5]	[ReLU, tansig]	BR	0.9453	NaN	1.6344	0.9985	100	60%, 20%, 20%
16	[10, 5]	[tansig, tansig]	LM	0.0237	1.2761	0.8583	0.9995	46	60%, 20%, 20%
17	[20 15]	[tansig, tansig]	LM	0.0014	1.613	1.4815	0.9992	13	60%, 20%, 20%
18	**[25 25]**	**[tansig, tansig]**	**LM**	$\mathbf{5.60 \times 10^{-3}}$	**0.7645**	**0.5519**	**0.9996**	**10**	**60%,20%,20%**
19	[15 22]	[logsig, logsig]	SCG	2.4642	2.3744	3.9888	0.9964	105	70%, 15%, 15%
20	[10 5 10]	[tansig, tansig, logsig]	LM	3.57×10^{-2}	2.0069	1.042	0.9994	31	70%, 15%, 15%

Table 5
Outsample testing points from FFBP (18 input) neural network structure.

Out-of-sample testing points from FFBP (18 input)					
Trial	AD (m)	MSE (m)	RMSE (m)	MAE (m)	R
1	1.0915	3.9826	1.9956	1.3012	0.9937
2	1.0864	4.5412	2.131	1.1306	0.99357
3	1.1627	4.9769	2.2309	1.3695	0.99244
4	**0.97448**	**2.9855**	**1.7279**	**1.1641**	**0.99572**
5	1.6707	8.6255	2.9369	1.8408	0.98685
6	1.0632	3.7107	1.9263	1.2042	0.99426
7	1.2311	4.8345	2.1988	1.4265	0.99289
8	1.2339	5.2326	2.2875	1.5254	0.99215
9	1.0418	3.6516	1.9109	1.2826	0.99449
10	1.2613	6.5464	2.5586	1.4911	0.99067
11	1.5064	6.9754	2.6411	1.763	0.98963
12	1.1654	4.4546	2.1106	1.2825	0.99346
13	1.2892	7.4662	2.7324	1.308	0.98865
14	1.1018	3.8617	1.9651	1.2387	0.99426
15	1.0229	3.1515	1.7753	1.0697	0.99534
16	1.0248	3.385	1.8398	1.0711	0.99493
17	1.195	5.4845	2.3419	1.3589	0.99177
18	**1.145**	**5.2814**	**2.2981**	**1.3475**	**0.99174**
19	1.1152	3.7534	1.9374	1.3051	0.99423
20	1.2429	4.6971	2.1673	1.2833	0.99275

Table 6
The performance criteria for the 20 input GRNN neural network structure.

Out-of-sample					
Spread Constant	AD (m)	MSE (m)	RMSE (m)	MAE (m)	R
0.5	0.60638	1.7215	1.3121	0.60638	0.9975

25% a testing set. The spread value was varied in the range of [0.01- 2] with a step distance of 0.01. Fig. 7 shows the effects of the spread parameter on the network's performance.

After the GRNN was trained using the optimal value of $\sigma = 0.5$, seventy testing points were used to verify its efficiency. Table 6 illustrates the different performance criteria of the tested structure.

(2) Input pattern based on a selective set of APs, which produce a high-level, acceptable RSSI and their coordinates.

This neural network structure depends on the APs that produce high-level, acceptable RSSIs of the corresponding RPs. Hence, in our case, as with the FFBP, the input is $[RSSI_{AP1}, X_{AP1}, Y_{AP1}, RSSI_{AP2}, X_{AP2}, Y_{AP2}, \ldots, RSSI_{AP6}, X_{AP6}, Y_{AP6}]^T$ and the output is $[XY]^T$.

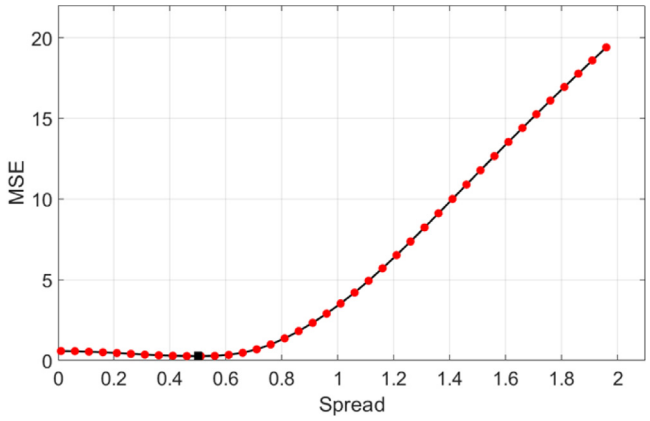


Fig. 7. The effect of spread parameter on the GRNN performance.

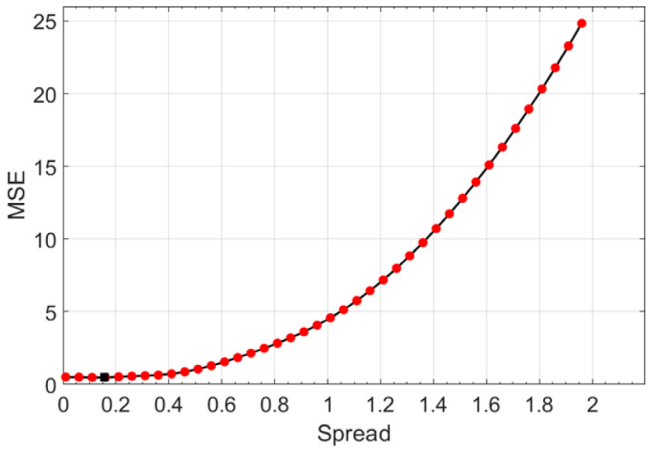


Fig. 8. Spread parameter.

Table 7
The performance criteria for the 18 input GRNN neural network structure.

Out-of-sample					
Spread Constant	AD (m)	MSE (m)	RMSE (m)	MAE (m)	R
0.16	0.48374	0.74249	0.86168	0.48374	0.9989

The split-sample cross-validation method is also used in this model to determine the spread constant's (σ) value. In this case, the optimal value of σ is 0.16, which corresponds with the minimum MSE value. Fig. 8 shows that the spread constant corresponds with the network's performance.

After the GRNN was trained using the optimal σ value, seventy testing points were used to verify its efficiency. Table 7 illustrates different performance criteria of the tested structure.

5. Results and discussion

The indoor location estimation system proposed in this paper is considered efficient in terms of time and efforts such that the interpolation leads to a great delay reduction from some hours to just a single run such that the data collection process for the real measurements needs approximately five hours of working time for RSSI values corresponds to only 64 reference points and 20 testing points. However, by the interpolated surfaces, a huge number of simulated measurements can be gathered in just a single run. In addition. A comparative analysis was conducted to select the most efficient neural network model. In the previous sections, the two proposed structures for FFBP network models achieved good performance for the in-sample and out-of-sample datasets. Table 8 shows a comparison between the two introduced FFBP network structures.

Table 8
Comparison between the two introduced FFBP structures.

Input Description	Input Neurons	Hidden Neurons	Activation Function	Training Algorithm	AD Error Out-of-sample (m)
RSSI from 20 APs	20	20	Tansig	LM	0.96662
RSSI from 6 strong APsand their coordinates	18	15	Tansig	LM	0.97448

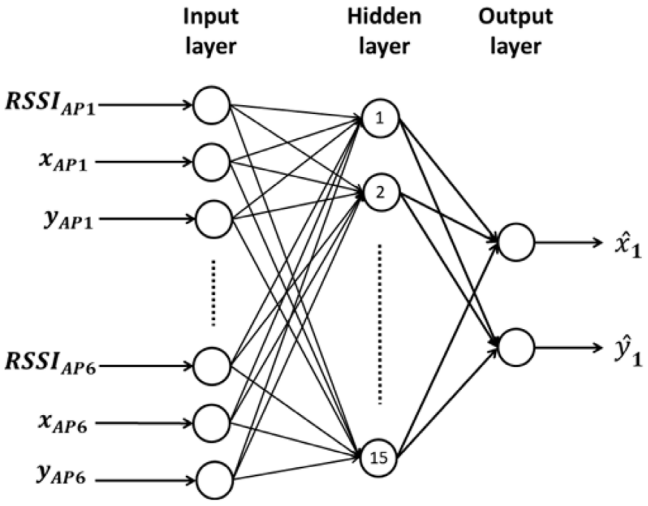


Fig. 9. FFBP neural network performance.

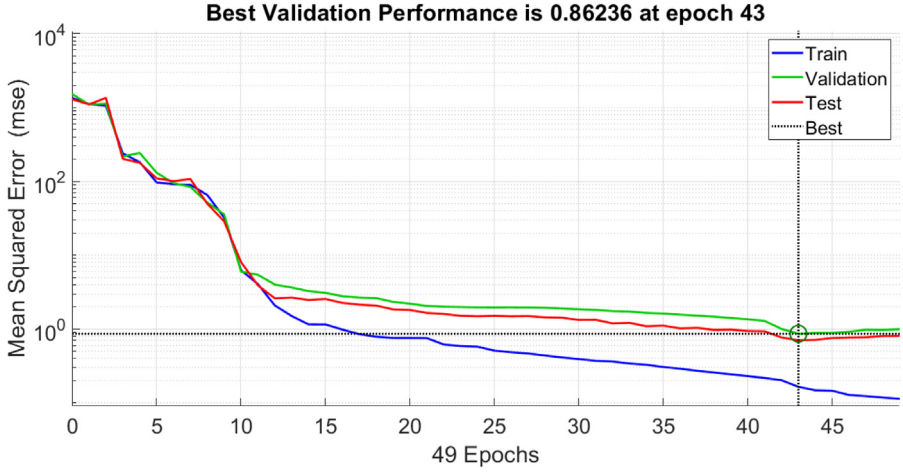


Fig. 10. FFBP neural network performance.

As shown in Table 8, both structures achieve good performance; however, the structure of the FFBP network model, which depends on the selective set of APs, is simpler than the other FFBP structure. The selected FFBP network model is shown in Fig. 9.

During the learning process, the dataset was divided into training (134), validation (28) and testing samples (28). The first two sets were used during the training phase, and the last set was used to evaluate the training process. Using the early-stop technique and the validation set, overfitting was reduced. The MSE versus epochs comparison of the trained networks is shown in Fig. 10. Fig. 11 shows the correlation between the predicted and actual results and for the training, validation and testing sets.

Table 9 shows a brief comparison of the results obtained from the two proposed GRNN models.

As illustrated in Table 9, the main difference between the two is in AD — the GRNN structure depends on a selective set of APs that have better performance. Hence, the selected GRNN model is shown in Fig. 12.

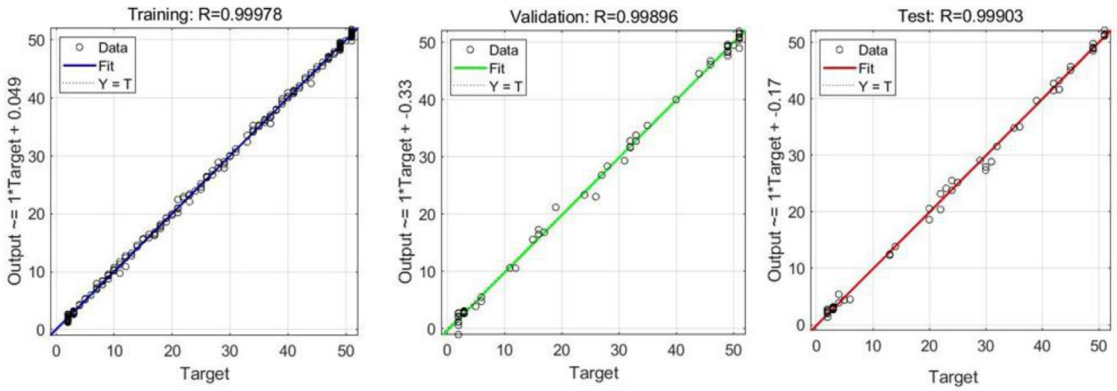


Fig. 11. Regression of error for FFBP neural network.

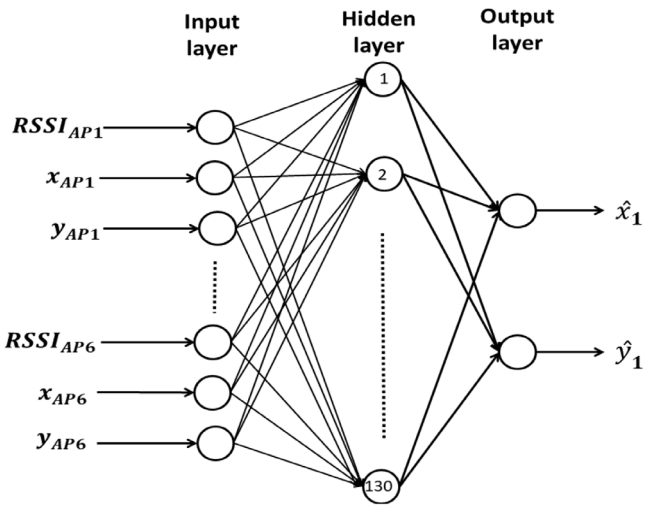


Fig. 12. Proposed GRNN structure.

Table 9
Comparison between GRNN models.

Input Description	Input Neurons	Hidden Neurons	Spread Constant	AD Error Out-of-sample (m)
RSSI from 20 APs	20	130	0.5	0.60638
RSSI from 6 strong APsand their coordinates	18	130	0.16	0.48374

Table 10
Comparison between FFBP and GRNN.

Neural Network	Input Neurons	Hidden Neurons	Activation Function	Training Algorithm	Spread Constant	AD Error Out-Of-sample (m)
FFBP	18	15	tansig	LM	–	0.97448
GRNN	18	130	–	–	0.16	0.48374

During the learning process, the dataset was separated into two sets, training and testing. Fig. 13 shows the correlation between the actual and predicted values.

As a summary, as shown in Table 10, the results show that FFBP outperforms GRNN in terms of structure simplicity, while GRNN achieved more accurate prediction results with an average distance error up to 0.48 m.

Furthermore, Fig. 14 shows the AD, for both FFBP and GRNN, of 20 out-of-sample points. The figure illustrates that the predicted locations using GRNN generally have AD values that are lower than the locations predicted by FFBP, such that in the GRNN model, nine testing points have zero AD, six have an AD of less than 0.5 m and the other five have an AD of less than 1.5 m. In the FFBP model, five testing points have an AD of less than 0.5 m, 10 have an AD of less than 1.5 m and the remaining five are 1.5–2.3 m.

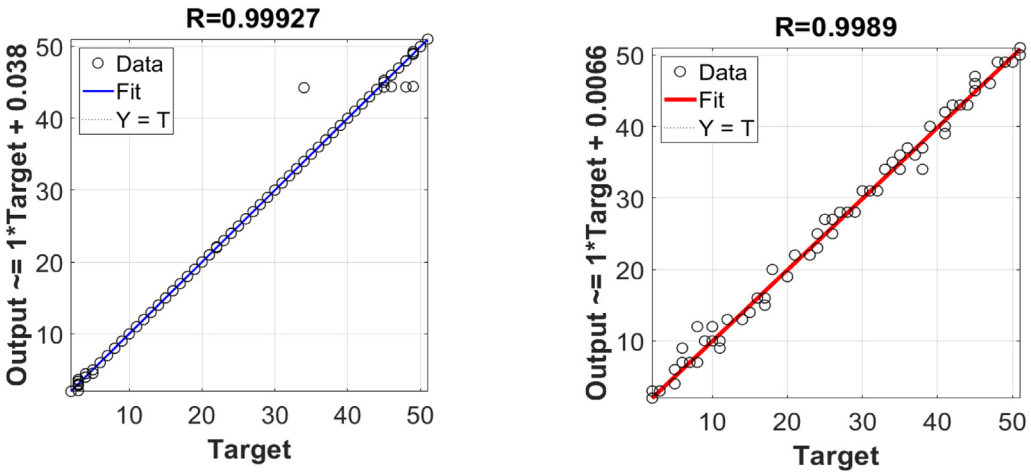


Fig. 13. Correlation between actual and predicted data for the proposed GRNN model(a) for training dataset (b) for testing dataset.

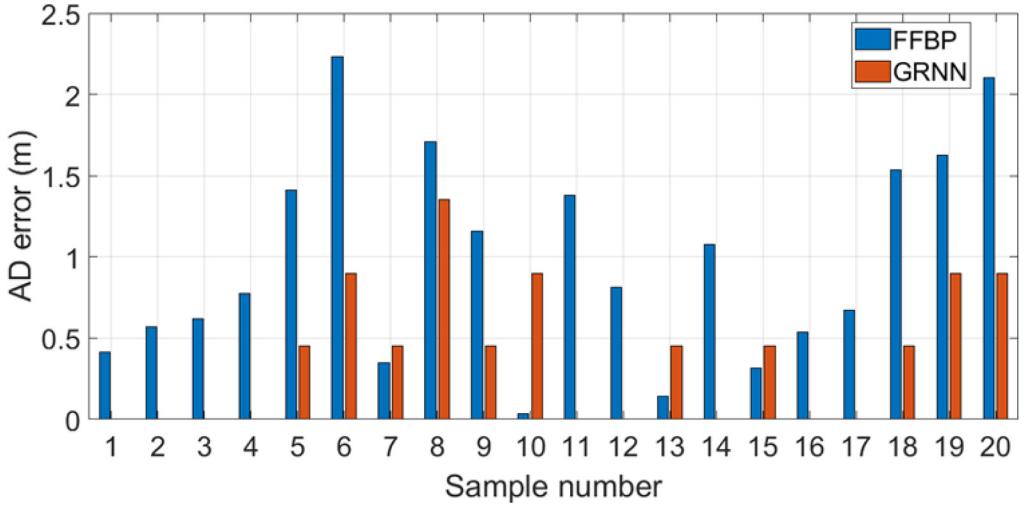


Fig. 14. AD error for FFBP and GRNN neural network.

In this paper, the positioning accuracy increases due to the efficient selection of the neural network model besides using dual-band APs, because using both 2.4 and 5 GHz frequencies provides a more accurate positioning [78]. A comparison between our approach and the different methods used in the state-of-the-art studies is conducted to justify the superiority of our proposed IPS with the obtained localisation results

- Study #1:
  The crowdsourcing method is used in [22] to build the initial radio map, which may have poor performance due to device heterogeneity problems that result in inaccurate locations since multiple users used various devices to perform the data collection process in the offline phase. The author claims that this problem is avoided with normalised RSSI fingerprints for all RPs and 58 APs are deployed in 3200 m^2. Their results show that the accuracy of the proposed method is RMSE = 2.938 m; in our study, one of the proposed ANN models is used just six APs, and the proposed models achieve better accuracy: RMSE = 1.589 for the FFNN and RMSE= 0.835 for the GRNN model.
- Study #2:
  The author of [24] tried to reduce the computational effort needed for the Htrack map-matching system and achieved localisation accuracy of RMSE $\approx$ 4 m and 3.5 for the museum and the office building experiment, respectively. The Wi-Fi fingerprinting method has a low computational requirement [79], and our obtained localisation results are more accurate.

- Study #3:
  In [21], the BSI method was used to generate a denser radio map and the KNN method for the location estimation process. However, for the KNN method, the output completely relies on the nearest neighbours. That is, it is sensitive to the noise and the multipath loss [80,81]; their resulting average position accuracy was MAE $\approx$ 1 m.
- Study #4:
  In [34], an IPS is executed in a museum environment based on Bluetooth Low Energy (BLE), the location estimation process is done with a feedforward neural network, and their results give a position accuracy below 1 m which is lower than our achieved accuracy. Bluetooth has some restrictions because it requires an additional installation while WiFi-based systems have no need to create an additional environment.

Compared to the previously mentioned works, our proposed IPS is a comprehensive system that handles several issues and performs better in terms of localisation accuracy, simplicity and effort.

In addition to the above comparison, the proposed model has the potential to be implemented in large-scale heterogeneous practical environments. Machine learning algorithms, particularly ANNs, are highly effective techniques that are able to overcome the limitations of traditional indoor positioning algorithms, which suffer from a lack of scalability, which adversely affects their performance in heterogeneous environments. On the other hand, neural network models, as mentioned in the literature, are very flexible in adapting well to dynamically changing environments, as well as multidimensional and heterogeneous data applications [82]. Mehmood in [83], discussed the efficiency of using ANN in heterogeneous indoor environments with different human activities, materials of walls, and types of APs. Moreover, the authors in [84], developed an ANN model for multi-building and multi-floor indoor localisation based on Wi-Fi fingerprinting. Consequently, the proposed system can operate efficiently at college campuses, hospitals, and airports since the Wi-Fi infrastructure and coverage is good. However, to achieve that objective, neural networks require more data. Therefore, as future work, we propose to develop a robot to implement a data collection process that will increase the usefulness of the proposed system.

6. Conclusion

In this work, we propose a WiFi-fingerprinting localisation system, which is not only able to save time and effort for radio-map establishment, but also achieves excellent indoor localisation performance. In the offline stage, a mobile application is built to gather RSSI fingerprints and construct a stored dataset, which is then uses the biharmonic spline interpolation method to generate a denser database. In the online stage, two different types of ANN are introduced for the location estimation process; these are FFBP neural networks and GRNN. For each type of neural network, different input patterns have been compared.

Several training simulations were carried out with different evaluation criteria. Both neural networks have shown decent modelling performance; however, the results show that FFBP outperforms GRNN in terms of structure simplicity while GRNN achieved more accurate prediction results with average distance error of up to 0.48 m.

Future work can be carried out using a more accurate method for the data-collection stage. A robot can be built to gather actual data, which can save more time and effort in the offline phase and guarantee high accuracy, which will affect the location-estimation process in the online phase.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

We acknowledge the support of An-Najah National University, Palestine.

Funding

This research received no external funding.

References

[1] E. Shakshuki, A.A. Elkhail, I. Nemer, M. Adam, T. Sheltami, Comparative study on range free localization algorithms, Procedia Comput. Sci. 151 (2019) 501–510.
[2] A. Al-baidhani, Self-deployable positioning systems for emergency situations employing uwb radio technology, 2019.
[3] W. Yuan, N. Wu, B. Etzlinger, H. Wang, J. Kuang, Cooperative joint localization and clock synchronization based on gaussian message passing in asynchronous wireless networks, IEEE Trans. Veh. Technol. 65 (9) (2016) 7258–7273.
[4] L. Gui, Improvement of Range-Free Localization Systems in Wireless Sensor Networks, (Ph.D. thesis), INSA, Toulouse, 2013.
[5] Y. Xiong, N. Wu, Y. Shen, M.Z. Win, Cooperative network synchronization: Asymptotic analysis, IEEE Trans. Signal Process. 66 (3) (2017) 757–772.

[6] S. Jie, D. En-qing, Z. Zong-jun, Y. Yuan, H. Zhen-qiang, A time synchronization protocol for large scale wireless sensor networks, in: IEICE Proceedings Series, Vol. 28, No. 15-PM1-D-3, 2015.
[7] A. Boukerche, D. Turgut, Secure time synchronization protocols for wireless sensor networks, IEEE Wirel. Commun. 14 (5) (2007) 64–69.
[8] W. Yuan, N. Wu, Q. Guo, X. Huang, Y. Li, L. Hanzo, Toa-based passive localization constructed over factor graphs: A unified framework, IEEE Trans. Commun. 67 (10) (2019) 6952–6965.
[9] K.S. Kim, R. Wang, Z. Zhong, Z. Tan, H. Song, J. Cha, S. Lee, Large-scale location- aware services in access: Hierarchical building/floor classification and location estimation using wi-fi fingerprinting based on deep neural networks, Fiber Integr. Opt. 37 (5) (2018) 277–289.
[10] M. Nowicki, J. Wietrzykowski, Low-effort place recognition with WiFi fingerprints using deep learning, in: International Conference Automation, Springer, 2017, pp. 575–584.
[11] A.B. Adege, H.-P. Lin, G.B. Tarekegn, S.-S. Jeng, Applying deep neural network (dnn) for robust indoor localization in multi-building environment, Appl. Sci. 8 (7) (2018) 1062.
[12] C. Rizos, A.G. Dempster, B. Li, J. Salter, Indoor positioning techniques based on wireless lan, 2007.
[13] R. Battiti, A. Villani, T. Le Nhat, Neural network models for intelligent networks: deriving the location from signal patterns, in: Proceedings of AINS, 2002.
[14] J. Jiao, F. Li, Z. Deng, W. Ma, A smartphone camera-based indoor positioning algorithm of crowded scenarios with the assistance of deep cnn, Sensors 17 (4) (2017) 704.
[15] L. Wu, C.-H. Chen, Q. Zhang, A mobile positioning method based on deep learning techniques, Electronics 8 (1) (2019) 59.
[16] M.H. Beale, M.T. Hagan, H.B. Demuth, Neural Network Toolbox User's Guide, The Mathworks Inc, 1992.
[17] An-Najah National University, 2020, Available online: https://www.najah.edu/en/ (Accessed 15 December 2020).
[18] Z. Sun, R. Farley, T. Kaleas, J. Ellis, K. Chikkappa, Cortina: Collaborative context- aware indoor positioning employing RSS and RToF techniques, in: 2011 IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Work- shops, IEEE, 2011, pp. 340–343.
[19] L. Chen, B. Li, K. Zhao, C. Rizos, Z. Zheng, An improved algorithm to generate awi-fi fingerprint database for indoor positioning, Sensors 13 (8) (2013) 11085–11096.
[20] A. Narzullaev, Y. Park, K. Yoo, J. Yu, A fast and accurate calibration algorithm for real-time locating systems based on the received signal strength indication, AEU- Int. J. Electron. Commun. 65 (4) (2011) 305–311.
[21] B. Li, Y. Wang, H.K. Lee, A. Dempster, C. Rizos, Method for yielding a database of location fingerprints in wlan, IEE Proc. Commun. 152 (5) (2005) 580–586.
[22] J. Bi, Y. Wang, H. Cao, H. Qi, K. Liu, S. Xu, A method of radio map construction based on crowdsourcing and interpolation for wi-fi positioning system, in: 2018 International Conference on Indoor Positioning and Indoor Navigation, IPIN, IEEE, 2018, pp. 1–6.
[23] B. Ferris, D. Fox, N.D. Lawrence, Wifi-slam using gaussian process latent variablemodels, in: IJCAI, Vol. 7, 2007, pp. 2480–2485.
[24] Y. Wu, P. Chen, F. Gu, X. Zheng, J. Shang, Htrack: An efficient heading-aidedmap matching for indoor localization and tracking, IEEE Sens. J. 19 (8) (2019) 3100–3110.
[25] D. Halim, A. Rusli, Wi-fi based indoor localization for location-based smart notification, IJNMT(Int. J. New Media Technol.) 7 (1) (2020) 43–50.
[26] R.C. Alves, J.S. de Morais, K. Yamanaka, Cost-effective indoor localization for autonomous robotsusing kinect and wifi sensors, Inteligencia Artif. 23 (65) (2020) 33–55.
[27] A. Ravi, A. Misra, Practical server-side wifi-based indoor localization: Addressing cardinality & outlierchallenges for improved occupancy estimation, Ad Hoc Netw. 115 (2021) 102443.
[28] H.X. Jian, W. Hao, Wifi indoor location optimization method based on position fingerprint algorithm, in: Proceedings - 2017 International Conference on Smart Grid and Electrical Automation, ICSGEA 2017, 2017, http://dx.doi.org/10.1109/ICSGEA.2017.123.
[29] S. Liu, J. Chai, Research of location fingerprint based on three-dimensional indoor positioning system, in: Proceedings - 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics, CISP-BMEI 2016, 2017, http://dx.doi.org/10.1109/CISP-BMEI.2016.7852882.
[30] E. Teoman, T. Ovatman, Trilateration in indoor positioning with an uncertain reference point, in: Proceedings of the 2019 IEEE 16th International Conference on Networking, Sensing and Control, ICNSC 2019, 2019, http://dx.doi.org/10.1109/ICNSC.2019.8743240.
[31] I.V. Korogodir, V.V. Dneprov, O.K. Mikhaylova, Triangulation positioning by means of Wi-Fi signals in indoor conditions, in: Progress in Electromagnetics Research Symposium, 2019, http://dx.doi.org/10.1109/PIERS-Spring46901.2019.9017863.
[32] E. Saad, M. Elhosseini, A.Y. Haikal, Recent achievements in sensor localizationalgorithms, Alex. Eng. J. 57 (4) (2018) 4219–4228.
[33] Y.P. Weerasinghe, M. Maduranga, M. Dissanayake, Rssi and feed forward neural network (ffnn) based indoor localization in wsn, in: 2019 National Information Technology Conference, NITC, IEEE, 2019, pp. 35–40.
[34] R. Giuliano, G.C. Cardarilli, C. Cesarini, L. Di Nunzio, F. Fallucchi, R. Fazzolari, F. Mazzenga, M. Re, A. Vizzarri, Indoor localization system based on bluetooth low energy for museum applications, Electronics 9 (6) (2020) 1055.
[35] X. Zhang, Y. Qiao, F. Meng, C. Fan, M. Zhang, Identification of maize leaf diseasesusing improved deep convolutional neural networks, IEEE Access 6 (2018) 30370–30377.
[36] G. Ding, Z. Tan, J. Zhang, L. Zhang, Fingerprinting localization based on affinity propagation clustering and artificial neural networks, in: 2013 IEEE Wireless Communications and Networking Conference, WCNC, IEEE, 2013, pp. 2317–2322.
[37] W. Zhang, K. Liu, W. Zhang, Y. Zhang, J. Gu, Deep neural networks for wireless localization in indoor and outdoor environments, Neurocomputing 194 (2016) 279–287.
[38] J.-W. Jang, S.-N. Hong, Indoor localization with wifi fingerprinting using convolutional neural network, in: 2018 Tenth International Conference on Ubiquitous and Future Networks, ICUFN, IEEE, 2018, pp. 753–758.
[39] S.Y.M. Vaghefi, R.M. Vaghefi, A novel multilayer neural network model for toa-based localization in wireless sensor networks, in: The 2011 International Joint Conference on Neural Networks, IEEE, 2011, pp. 3079–3084.
[40] Y.-K. Cheng, H.-J. Chou, R.Y. Chang, Machine-learning indoor localization with access point selection and signal strength reconstruction, in: 2016 IEEE 83rd Vehicular Technology Conference, VTC Spring, IEEE, 2016, pp. 1–5.
[41] J. Yoo, K.H. Johansson, H.J. Kim, Indoor localization without a prior map by trajectory learning from crowdsourced measurements, IEEE Trans. Instrum. Meas. 66 (2017) 2825–2835.
[42] Y. Fu, P. Chen, S. Yang, J. Tang, An indoor localization algorithm based on continuous feature scaling and outlier deleting, IEEE Internet Things J. 5 (2018) 1108–1115.
[43] X. Guo, N. Ansari, L. Li, H. Li, Indoor localization by fusing a group of fingerprints based on random forests, IEEE Internet Things J. (2018).
[44] Haeberlen A., E. Flannery, A.M. Ladd, A. Rudys, D.S. Wallach, L.E. Kavraki, Practical robust localization over large-scale 802.11 wireless networks, in: Proceedings of the MobiCom'04, Philadelphia, PA, USA, 26 September–1, 2004.
[45] S.K. Gharghan, R. Nordin, A.M. Jawad, H.M. Jawad, M. Ismail, Adaptive neural fuzzy inference system for accurate localization of wireless sensor network in outdoor and indoor cycling applications, IEEE Access 6 (2018) 38475–38489.
[46] J. Jang, S. Hong, Indoor localization with WiFi fingerprinting using convolutional neural network, in: Proceedings of the 2018 Tenth International Conference on Ubiquitous and Future Networks, ICUFN, Prague, Czech Republic, 2018, pp. 53–758, 3–6 2018.

[47] M. Ibrahim, M. Torki, M. ElNainay, CNN based indoor localization using RSS time-series, in: Proceedings of the 2018 IEEE Symposium on Computers and Communications, ISCC, Natal, Brazil, 2018, pp. 01044–01049, 25–28 2018.
[48] X. Wang, X. Wang, S. Mao, Deep convolutional neural networks for indoor localization with CSI images, IEEE Trans. Netw. Sci. Eng. (2018).
[49] A. Niitsoo, T. Edelhauber, C. Mutschler, Convolutional neural networks for position estimation in TDoA-based locating systems, in: Proceedings of the 2018 International Conference on Indoor Positioning and Indoor Navigation, IPIN, Nantes, France, 2018, pp. 1–8, 24–27 2018.
[50] D. Konings, B. Parr, F. Alam, E.M. Lai, Falcon: Fused application of light based positioning coupled with onboard network localization, IEEE Access 6 (2018) 36155–36167.
[51] X. Zhang, H. Sun, S. Wang, J. Xu, A new regional localization method for indoor sound source based on convolutional neural networks, IEEE Access 6 (2018) 72073–72082.
[52] N. Anzum, S.F. Afroze, A. Rahman, Zone-based indoor localization using neural networks: A view from a real testbed, in: Proceedings of the 2018 IEEE International Conference on Communications, ICC, Kansas City, MO, USA, 2018, pp. 1–7, 20-24 2018.
[53] A.B. Adege, L. Yen, H.P. Lin, Y. Yayeh, Y.R. Li, S.S. Jeng, G. Berie, Applying deep neural network (DNN) for large-scale indoor localization using feed-forward neural network (FFNN) algorithm, in: Proceedings of the 2018 IEEE International Conference on Applied System Invention, ICASI, Chiba, Japan, 2018, pp. 814–817, 13–17 2018.
[54] E.L. Berz, D.A. Tesch, F.P. Hessel, Machine-learning-based system for multi-sensor 3D localisation of stationary objects, IET Cyber Phys. Syst. Theory Appl. 3 (2018) 81–88.
[55] C. Zhu, L. Xu, X. Liu, F. Qian, Tensor-generative adversarial network with two-dimensional sparse coding: Application to real-time indoor localization, in: Proceedings of the 2018 IEEE International Conference on Communications, ICC, Kansas City, MO, USA, 2018, pp. 1–6, 20–24 2018.
[56] J. Li, Y. Wei, M. Wang, J. Luo, Y. Hu, Two indoor location algorithms based on sparse fingerprint library, in: Proceedings of the 2018 Chinese Control and Decision Conference, CCDC, Shenyang, China, 2018, 9–11 2018.
[57] J. Choi, W. Lee, J. Lee, J. Lee, S. Kim, Deep learning based NLOS identification with commodity WLAN devices, IEEE Trans. Veh. Technol. 67 (2018) 3295–3303.
[58] B.A. Akram, A.H. Akbar, O. Shafiq, HybLoc: Hybrid IndoorWi-Fi localization using soft clustering-based random decision forest ensembles, IEEE Access 6 (2018) 38251–38272.
[59] Z. Huang, J. Xu, J. Pan, A regression approach to speech source localization exploiting deep neural network, in: Proceedings of the 2018 IEEE Fourth International Conference on Multimedia Big Data, BigMM, Xi'an, China, 2018, pp. 1–6, 13–16 2018.
[60] S. He, S.-H.G. Chan, Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons, IEEE Commun. Surv. Tutor. 18 (1) (2015) 466–490.
[61] J. Du, J.-F. Diouris, Y. Wang, A rssi-based parameter tracking strategy for con- strained position localization, EURASIP J. Adv. Signal Process. 2017 (1) (2017) 1–10.
[62] J. Racko, J. Machaj, P. Brida, Wi-fi fingerprint radio map creation by using interpolation, Procedia Eng. 192 (2017) 753–758.
[63] R. Kubota, S. Tagashira, Y. Arakawa, T. Kitasuka, A. Fukuda, Efficient surveydatabase construction using location fingerprinting interpolation, in: 2013 IEEE 27thinternational Conference on Advanced Information Networking and Applications, AINA, IEEE, 2013, pp. 469–476.
[64] L.I. Feliciano-Cruz, E.I. Ortiz-Rivera, Biharmonic spline interpolation for solar ra-diation mapping using puerto rico as a case of study, in: 2012 38th IEEE PhotovoltaicSpecialists Conference, IEEE, 2012, pp. 002913–002915.
[65] D.T. Sandwell, Biharmonic spline interpolation of geos-3 and SEASAT altimeter data, Geophys. Res. Lett. 14 (2) (1987) 139–142.
[66] Y. Yang, P. Dai, H. Huang, M. Wang, Y. Kuang, A semi-simulated RSS fingerprint construction for indoor wi-fi positioning, Electronics 9 (10) (2020) 1568.
[67] C. Nwankpa, W. Ijomah, A. Gachagan, S. Marshall, Activation functions: Comparison of trends in practice and research for deep learning, 2018, Preprint arXiv:1811.03378.
[68] L. Gong, C. Liu, Y. Li, Y. Fuqing, Training feed-forward neural networks using the gradient descent method with the optimal step size, J. Comput. Inform. Syst. 8 (4) (2012) 1359–1371.
[69] N. Gong, W. Shao, H. Xu, The conjugate gradient method with neural network control, in: 2010 IEEE International Conference on Intelligent Systems and Knowledge Engineering, IEEE, 2010, pp. 82–84.
[70] M.F. Møller, A scaled conjugate gradient algorithm for fast supervised learning, Neural Netw. 6 (4) (1993) 525–533.
[71] F. Burden, D. Winkler, BayesIan regularization of neural networks, in: Artificial Neural Networks, Springer, 2008, pp. 23–42.
[72] L. Ma, F. Xu, X. Wang, L. Tang, Earthquake prediction based on levenberg- marquardt algorithm constrained back-propagation neural network using demeter data, in: International Conference on Knowledge Science, Engineering and Management, Springer, 2010, pp. 591–596.
[73] S. Chen, C.F. Cowan, P.M. Grant, Orthogonal least squares learning algorithm for radial basis function networks, IEEE Trans. Neural Netw. 2 (2) (1991) 302–309.
[74] A.A. Konate, H. Pan, N. Khan, J.H. Yang, Generalized regression and feed-forward back propagation neural networks in modelling porosity from geophysical well logs, J. Petrol. Explor. Prod. Technol. 5 (2) (2015) 157–166.
[75] C.-H. Cheng, T.-P. Wang, Y.-F. Huang, Indoor positioning system using artificial neural network with swarm intelligence, IEEE Access 8 (2020) 84248–84257.
[76] R.P. Ghozali, G.P. Kusuma, Indoor positioning system using regression-based fingerprint method, Int. J. Adv. Comput. Sci. Appl. 10 (8) (2019).
[77] K.G. Sheela, S.N. Deepa, Review on methods to fix number of hidden neurons in neural networks, Math. Probl. Eng. (2013).
[78] F. Karlsson, M. Karlsson, B. Bernhardsson, F. Tufvesson, M. Persson, Sensor fused indoor positioning using dual band wifi signal measurements, in: 2015 European Control Conference, ECC, IEEE, 2015, pp. 1669–1672.
[79] W. Wang, D. Marelli, M. Fu, Fingerprinting-based indoor localization using interpolated preprocessed csi phases and bayesian tracking, Sensors 20 (10) (2020) 2854.
[80] W. Hongpeng, F. Jia, A hybrid modeling for wlan positioning system, in: 2007 in-International Conference on Wireless Communications, Networking and Mobile Computing, IEEE, 2007, pp. 2152–2155.
[81] Y. Liu, R.S. Sinha, S.-Z. Liu, S.-H. Hwang, Side-information-aided preprocessing scheme for deep-learning classifier in fingerprint-based indoor positioning, Electronics 9 (6) (2020) 982.
[82] A. Nessa, B. Adhikari, F. Hussain, X.N. Fernando, A survey of machine learning for indoor positioning, IEEE Access 8 (2020) 214945–214965.
[83] H. Mehmood, N.K. Tripathi, T. Tipdecho, Indoor positioning system using artificial neural network, J. Comput. Sci. 6 (10) (2010) 1219.
[84] K.S. Kim, S. Lee, K. Huang, A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on wi-fi fingerprinting, Big Data Anal. 3 (1) (2018) 1–17.