IEC
Institute of Electronics
and Computer

Journal of Artificial Intelligence and Systems, 2026, 8, 1–17
https://iecscience.org/journals/AIS
ISSN Online: 2642–2859

Quantum Neural Regressors in Localization Tasks for Wireless Communications Networks

Vinicius R. de Albuquerque[1], **Fernando M. de Paula Neto**[2,*], **Daniel C. Cunha**[3]

[1]Centro de Informática, Universidade Federal de Pernambuco, Recife, Pernambuco, Brazil
Email: vra2@cin.ufpe.br
[2]Centro de Informática, Universidade Federal de Pernambuco, Recife, Pernambuco, Brazil
Email: dcunha@cin.ufpe.br
[3]Centro de Informática, Universidade Federal de Pernambuco, Recife, Pernambuco, Brazil
Email: fernando@cin.ufpe.br
*Corresponding Author: Fernando M. de Paula Neto, Email: fernando@cin.ufpe.br

How to cite this paper: Vinicius R. de Albuquerque, Fernando M. de Paula Neto and Daniel C. Cunha (2026) Quantum Neural Regressors in Localization Tasks for Wireless Communications Networks. Journal of Artificial Intelligence and Systems, 8, 1–17.
https://doi.org/10.33969/AIS.2026.080101

Received: May 11, 2026
Accepted: July 20, 2026
Published: July 31, 2026

Abstract

In this work, we applied two quantum neural circuit (QNC) models as regressors for the average localization error prediction problem in a wireless sensor network and for mobile phone position estimation using a wireless local area network (WLAN)- fingerprinting-based positioning strategy. We trained the QNCs using the variational circuit strategy with three different types of optimizers. The results were compared with six classical regressors widely used in the literature: linear regressor (LR), support vector regressor (SVR), multilayer perceptron regressor (MLPR), decision tree regressor (DTR), K-nearest neighbors regressor (KNNR), and stochastic gradient descent regressor (SGDR). Different configurations of the SVR and MLPR models were tested. The results showed that quantum circuit optimization was promising for these localization tasks and that a QNC model (i.e., a single quantum neuron) was statistically equivalent to, or even superior to, the six classical regressor models in many cases (according to the Wilcoxon test). The results obtained in this work are promising for quantum computing, particularly as small-scale quantum systems become increasingly common.

Keywords

Quantum Neural Networks, Quantum Machine Learning, Wireless Sensor Networks, Indoor Localization.

1. Introduction

Quantum computing has emerged as a promising paradigm for solving computationally complex problems, exemplified by Shor's algorithm for integer factorization [1]. This innovation, along with other quantum algorithms for optimization, simulation, and search, has established the transformative potential of quantum technologies. More recently, these advances have been extended to the field of quantum machine learning (QML) [2, 3], which uses quantum resources, such as superposition and entanglement, to improve computational models. In QML, quantum neuron architectures have been proposed, demonstrating expressive power for various nonlinear tasks and serving as quantum analogs of classical neurons and neural networks [4–7].

Recent studies have further strengthened the theoretical and empirical foundations of quantum neural models. In [8], it is argued that quantum neural networks can outperform classical convolutional neural networks under certain conditions. Additionally, [9] shows that there exist feature maps and quantum kernels that enable variational quantum classifiers and quantum kernel support vector machines to efficiently solve any problem in the complexity BQP class[1]. This result suggests that properly designed quantum feature maps may provide quantum advantage for intractable classification tasks.

From a theoretical perspective, the expressive power of variational quantum circuits (VQCs) has been extensively investigated. In [10], it is shown that such models can universally approximate functions through Fourier series induced by data encoding strategies. Similarly, [11] establishes universality results based on the universal approximation theorem. Despite this strong expressivity, designing suitable quantum circuits for specific problems remains a significant challenge, as there is no general method for constructing optimal VQCs. However, recent works indicate that automated machine learning approaches may assist in this process [12, 13]. On the other hand, simpler quantum neuron models [6, 7] stand out for their reduced circuit complexity and limited gate sets, but they have not yet been extensively evaluated across arbitrary and complex problem domains. Furthermore, it is necessary to assess the training quality of VQC models, as although they can be trained, they can converge to undesirable local minima, hindering proper model optimization [14].

Machine learning (ML) is crucial to achieve precise localization in modern wireless communication networks, including wireless sensor networks (WSNs) and mobile positioning systems [15–18]. For WSNs, minimizing the average localization error (ALE) is a significant challenge. Traditional optimization techniques often face limitations, such as premature convergence to local minima and high computational costs, especially in dynamic network configurations where parameters require frequent retuning [19–21]. Although classical ML approaches can approximate ALE predictions, they may struggle with the inherent complexity and nonlinearity of sensor data [22]. Similarly, Wi-Fi fingerprinting, while cost-effective for indoor localization, suffers from accuracy issues due to sensor errors, noise, and multipath effects. Additionally, maintaining radio maps can be challenging, as they require frequent updates to account for environmental changes [23, 24]. These persistent challenges in both WSNs and indoor Wi-Fi localization, particularly with high-dimensional, non-linear data, motivate the exploration of quantum-computational solutions [25–27].

This work introduces a novel quantum regression framework that distinguishes itself from previous QML efforts, which have focused primarily on classification [28] or on hybrid deep learning architectures [29, 30]. We utilize compact VQCs based on parameterized quantum neurons to directly predict continuous-valued localization metrics, such as ALE and user positions. This approach aims to address the aforementioned complexities by offering a standalone quantum regressor with significantly fewer parameters than traditional complex models.

We evaluate two quantum neuron models within a VQC training framework across two practical regression tasks: (i) predicting ALE in WSNs and (ii) estimating indoor user positions using Wi-Fi fingerprinting. By assessing these models with multiple optimizers and datasets, our study offers new insights into the expressive power and statistical behavior of quantum regressors in addressing real-world localization challenges.

The remainder of this paper is structured as follows: Section 2 reviews quantum computing concepts, while Section 3 presents the quantum neuron models. Section 4 details the experimental methodology. Section 5 presents the numerical results, and conclusions are drawn in Section 6.

[1] bounded-error quantum polynomial time class

2. Quantum Computing

A quantum bit, shortly named as a qubit, is a complex bidimensional unit vector. Vectors in quantum computing are usually denoted using the bra-ket notation, respectively $\langle\cdot|$ and $|\cdot\rangle$, proposed by Dirac [31]. Although qubits can be represented as linear combinations of any orthogonal basis of $\mathbb{C}^2$, the most widely used basis is *canonical* (or *computational*) defined as the pair of vectors $|0\rangle = [1,0]^T$ and $|1\rangle = [0,1]^T$. A qubit $|\psi\rangle$ can be written as

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle\,, \tag{1}$$

where α and β are the state amplitudes of the quantum state, $\alpha,\beta \in \mathbb{C}$, obeying the normalization condition $|\alpha|^2 + |\beta|^2 = 1$.

Composite quantum systems are formed using the tensor product $|\phi\psi\rangle = |\phi\rangle \otimes |\psi\rangle$. Usually, in quantum computing, one works with 2^k-dimensional Hilbert spaces, where k is the number of qubits. A general k-qubits state $|\phi\rangle$ is represented as

$$|\phi\rangle = \sum_{j=0}^{2^k-1} \alpha_j |\mathbf{j}\rangle\,. \tag{2}$$

Every quantum state represented by $|\psi\rangle = [c_0, c_1, c_2, \ldots]^T$, has a dual bra, written as a row vector $\langle\psi| = [c_0^*, c_1^*, c_2^*, \ldots]$, where $i = \sqrt{-1}$ is the imaginary number, and the complex number $c_i = a + b\cdot i$, a and $b \in \mathbb{R}$, has the conjugate transpose $c_i^* = a - b\cdot i$.

A quantum system is altered by an operator or a set of operators that change the qubit amplitude values. A quantum operator $\mathbf{U}$ on n qubits is a complex unitary matrix $2^n \times 2^n$. A complex unitary matrix is a square matrix with complex-valued entries such that $\mathbf{U}^\dagger\mathbf{U} = \mathbf{U}\mathbf{U}^\dagger = \mathbf{I}$, where $\mathbf{U}^\dagger$ is the transposed matrix obtained from the conjugate entries of $\mathbf{U}$ and $\mathbf{I}$ is the identity operator, such that

$$\mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \begin{matrix} \mathbf{I}|0\rangle = |0\rangle \\ \mathbf{I}|1\rangle = |1\rangle \end{matrix}\,. \tag{3}$$

Some other operators over one qubit are the *not* operator $\mathbf{X}$ and the *Hadamard* operator $\mathbf{H}$, described, respectively, in Eqs. (4) and (5):

$$\mathbf{X} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \begin{matrix} \mathbf{X}|0\rangle = |1\rangle \\ \mathbf{X}|1\rangle = |0\rangle \end{matrix} \tag{4}$$

$$\mathbf{H} = \tfrac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \begin{matrix} \mathbf{H}|0\rangle = 1/\sqrt{2}(|0\rangle + |1\rangle) \\ \mathbf{H}|1\rangle = 1/\sqrt{2}(|0\rangle - |1\rangle) \end{matrix}\,. \tag{5}$$

The identity operator $\mathbf{I}$ outputs the input; the not operator $\mathbf{X}$ behaves as the classical NOT on the computational basis, while the Hadamard operator $\mathbf{H}$ generates a superposition of states.

Quantum operations can be represented graphically by quantum circuits. A quantum circuit combines operators applied to one or some set of qubits. Fig. 1 shows an example of a quantum circuit composed of a *not* operator $\mathbf{X}$ and two **CNOT** operators. In this representation, the qubits are wires, and the quantum operators are boxes. The **CNOT** is a two-qubit operator with a control and a target qubit. A filled circle represents the control qubit, and the symbol $\oplus$ indicates the target qubit. The **CNOT** operator works considering the value of the control qubit to apply the *not* operator $\mathbf{X}$ on the target qubit. If the control qubit is set to 1, the *not* operator $\mathbf{X}$ is applied to the target qubit. The execution flow, as in the classical case, is from left to right. Finally, the matrix representation of the **CNOT** operator on the computational basis is given by

$$\mathbf{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \begin{matrix} \mathbf{CNOT}|00\rangle = |00\rangle \\ \mathbf{CNOT}|01\rangle = |01\rangle \\ \mathbf{CNOT}|10\rangle = |11\rangle \\ \mathbf{CNOT}|11\rangle = |10\rangle \end{matrix}\,. \tag{6}$$

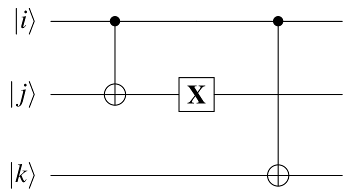

Figure 1. An example of quantum circuit with two **CNOT** operators and a *not* operator **X**.

Measurement in quantum mechanics is an irreversible process that results in the partial or complete loss of information regarding the superposition of states. Consider a qubit state $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$. When a measurement is performed, the state collapses (or projects) to either $|0\rangle$ with a probability of $|\alpha|^2$ or to $|1\rangle$ with a probability of $|\beta|^2$. For a composite and superposed quantum state $|\mathbf{g}\rangle$, the probability of observing a specific state $|\mathbf{j}\rangle$ is given by $|\langle\mathbf{g}|\mathbf{j}\rangle|^2$. An example of a measurement operator within a quantum circuit is illustrated in Fig. 2.

The expected value $\langle Z\rangle$ of this measurement for an arbitrary qubit $|\psi\rangle$ is expressed as $\langle Z\rangle = \langle\psi|\,Z\,|\psi\rangle$, where Z is the Pauli-Z matrix defined as

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

This expected value ranges between -1 and 1. The expectation value of an arbitrary observable $\mathbf{A}$, denoted as $\langle\mathbf{A}\rangle$, in the quantum initial state $|\psi\rangle$ evolved by the unitary operator $U(\theta)$ is shown in Eq. (7), such that

$$\langle\mathbf{A}\rangle = \langle\psi|\,U^{\dagger}(\theta)\mathbf{A}U(\theta)\,|\psi\rangle\,. \tag{7}$$

Measurement, or observation, is the practical way of extracting information from a quantum system. Only by measurement do we obtain a value for the quantum system, enabling us to use it to solve problems. For further information on quantum measurements, see [31].

3. Quantum Neural Regressors

The artificial neuron mimics the biological neuron, which receives one or more nerve impulse signals weighted by the number of receptors in each connection. The mathematical modeling of this interaction is traditionally seen as an inner product between a vector of input signals, $\theta = [\theta_1, \theta_2, \ldots, \theta_m]$, and a vector of weights, $\phi = [\phi_1, \phi_2, \ldots, \phi_m]$, which represents the significance factor of each signal, applied to a propagation function f, calculated by $\theta \cdot \phi = f(\sum_{j=1}^{m} \theta_j \phi_j)$ [32]. To allow for an additional degree of freedom in modeling this system, a constant input term and a weight are added to this input, enabling the system to be adjusted for different possible output threshold values. This weight value attached to a constant input is called bias, here mentioned as b, such that the final equation of operation of the neuron is given by $\theta \cdot \phi = f(\sum_{j=1}^{m} \theta_j \phi_j + b)$. The signal propagation function is usually nonlinear, allowing the output to represent even more complex information. There is a current effort to quantize classical operations, representing information and classical operations in a quantum model, which yields a gain under certain objective criteria [2].

In [6], a quantum neuron model named continuous-valued quantum neuron (CVQN) was proposed, in which the interaction operation of the input signal value with its weights is

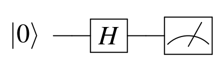

Figure 2. Measurement yielding 0 or 1 with equal probability $\frac{1}{2}$.

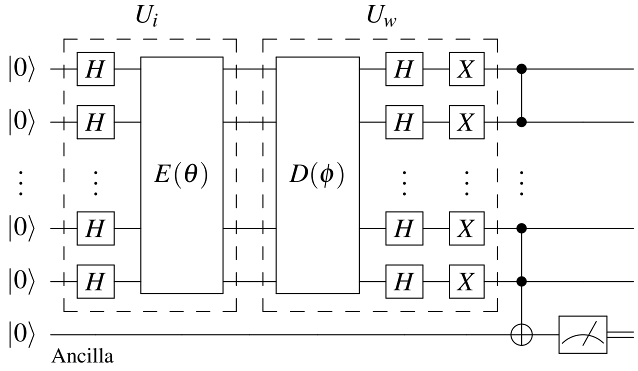

Figure 3. Framework circuit to build quantum neurons based on kernel machines.

first transformed into a new representation. After that, given m-dimensional classical input and weight vectors, denoted respectively as θ and ϕ, the quantum neuron is performed by the formula described as

$$\theta \cdot \phi = \left| \frac{1}{N} \sum_{j=1}^{m} e^{i(\theta_j - \phi_j)} \right|^2 + b \,, \tag{8}$$

where N is the normalization factor of the resulting quantum state and depends on the input size $N = 2^m$.

That work was extended in [7] to a generalized neuron architecture that expands the encoding of input values, enabling alternative mechanisms for information representation and weight modeling. In addition, the proposed formulation introduces new parameters that modify the interaction between inputs and weights, allowing the neuron to implement different interaction patterns and, consequently, distinct decision functions.

This generalized architecture is presented in Fig. 3. First, U_i encodes the input vector θ, using some set of quantum operators $E(\theta)$ described in terms of input θ. Then, U_w encodes the weight vector ϕ, using some set of quantum operators $D(\phi)$ described in terms of the weight ϕ, and computes the interaction between θ and ϕ. When finally measured, this information is extracted into an ancillary qubit, yielding a non-deterministic activation function.

By defining a specific configuration of this architecture, using $P(\lambda)$ gates, where $P(\lambda) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{pmatrix}$, it was possible to propose a parametric constant-depth quantum neuron (PCDQN) [7], which interacts with the input vector θ and weight vector ϕ, such that

$$\theta \cdot \phi = \left| \frac{1}{N} \prod_{j=1}^{m} (1 + e^{i[\tau(\theta_j - \phi_j) + \delta]}) \right|^2 + b \,. \tag{9}$$

Fig. 4 illustrates the PCDQN circuit. The hyperparameters of this model are the scalar values τ and δ, while the trainable parameters adjusted during the learning process are given by the vector θ.

In Fig. 5, it is possible to see the resume of the classical neuron formulas: the formulas of CVQN neuron, here called quantum circuit neuron (QCN), and PCDQN neuron, here called quantum parametrized neuron (QPN). From now on in the text, quantum neurons are referred to by these acronyms for the sake of simplicity of notation.

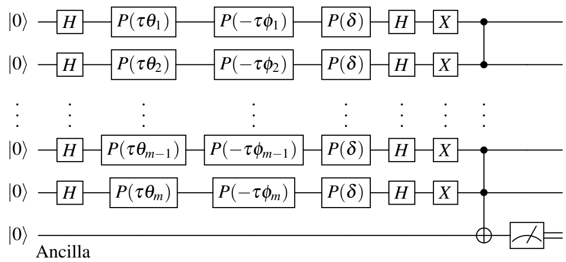

Figure 4. Circuit implementation of the parametric constant-depth quantum neuron (PCDQN). The parameters τ and δ can alter the shape of the PCDQN activation function.

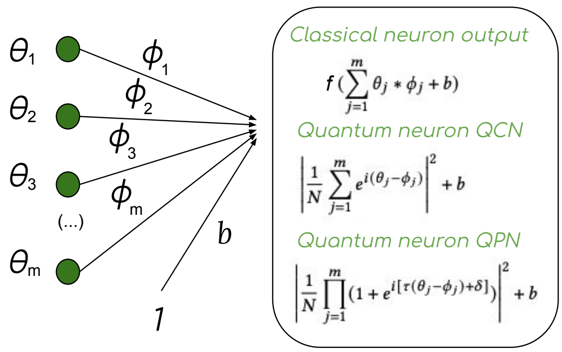

Figure 5. Classical and quantum activation functions for the neuron that processes m inputs $[\theta_1,\ldots,\theta_m]$, therefore, have m weights $[\phi_1,\ldots,\phi_m]$ and a bias value b. The value N is the normalization factor of the quantum state and is worth $N = 2^m$, and f is the activation function of the classical neuron and is generally a non-linear function.

3.1. Quantum Circuit Optimizers

The learning stage of a quantum circuit can be modeled by an iterative training via the delta rule update [33]. Each parameter is updated based on the previous state and the gradient of the error function. Gradient-based optimizers generally find a slightly better solution than gradient-free optimizers, and the adaptive learning rate or gradient strategy also improves the performance of hybrid quantum-classical models [34].

A cost function is defined to update a parameter value to find a minimum value for this function. This cost function evaluates the error of the circuit output, the average expected value measured using an arbitrary observable $\langle A\rangle$, with respect to the expected output value $\hat{y}$. The error calculation, denoted as $f(y,\langle A\rangle)$, defines the cost function $C(\theta)$ of the circuit for a given parameter value θ, so that

$$C(\theta) = f(\hat{y}, \langle\psi|U^{\dagger}(\theta)\mathbf{A}U(\theta)|\psi\rangle)\,. \tag{10}$$

The weight update formula of the gradient descent optimizer [35] is described as

$$\theta^{(t+1)} = \theta^{(t)} + \eta\nabla C(\theta^{(t)})\,, \tag{11}$$

where $\theta^{(t)}$ represents any circuit parameters at t-th iteration, η is the learning rate, and $\nabla C(\theta)$ is the cost function gradient in θ. Furthermore, we can accelerate the optimization's convergence rate by making the learning rate dependent on the previous step.

A few other optimizers exist as modifications of the traditional gradient descent optimizer [36]. The Nesterov Momentum optimizer is a gradient descent optimizer with Nesterov momentum. Nesterov Momentum works similarly to the Momentum optimizer, but shifts the current input by the momentum term when computing the gradient of the objective function. The Adam optimizer is a gradient-descent optimizer with an adaptive learning rate for both the first and second moments. Adaptive moment estimation uses a step-dependent learning rate, a first moment a, and a second moment b, reminiscent of the momentum and velocity of the update parameters. Adagrad is a gradient-descent optimizer with a learning rate that depends on past gradients in each dimension. The root mean squared (RMS) propagation optimizer is a modified Adagrad optimizer with a decay-based learning rate adaptation. The simultaneous perturbation stochastic approximation (SPSA) method is a stochastic approximation algorithm for optimizing cost functions that involve noise during evaluation. While other gradient-based optimization methods usually compute gradients analytically, SPSA approximates gradients at the cost of evaluating the cost function twice per iteration. This cost may significantly decrease the overall cost of function evaluations for the entire optimization. The learning rate of the optimizers is 0.01. The momentum is 0.9 for the Nesterov optimizer. For the Adam hyper-parameter optimizer, β_1 is assumed to be 0.9, while β_2 equals 0.99 [36].

4. Experimental Protocol

This section describes the experimental protocol adopted in this study. Two datasets were evaluated for each quantum neuron model — the QCN and the QPN — using the six optimizers discussed in Section 3.1. All experiments were implemented in Python (Version 3.11.5), employing the `Pennylane` library [36] (Version 0.42.3) for the construction and training of quantum circuits, and `scikit-learn` (Version 1.6.1) for the implementation of the classical regression baselines. The computations were performed on a workstation equipped with an Intel Core i7 processor and 16 GB of RAM.

In the subsections that follow, the two datasets used in the experiments will be described in detail, including their characteristics and preprocessing procedures. Briefly, the first dataset concerns the prediction of ALE in WSNs, while the second dataset, named *UJIIndoorLoc*, involves Wi-Fi fingerprint-based positioning. Their full descriptions and preprocessing pipelines are presented in the next subsections.

4.1. Dataset 1 (ALE in WSNs)

The first dataset focuses on the locations of nodes within a WSN. It consists of 107 different network configurations characterized by five key parameters: anchor ratio, transmission range, node density, Cuckoo Search iterations, and the ALE value for each configuration. Tab. 1 provides the details of this database.

During data preprocessing, all columns in the database were normalized to a scale of 0 to 1 using the min-max normalization method. For each column x, each value x_i was normalized as $x_i/\max(x)$. Using this dataset, we can train intelligent models capable of predicting the ALE from a given WSN configuration [22].

Table 1. Details of the wireless sensor network (WSN) database.

Feature	Range of values	Type
Node density	$[100, 200, 300]$	Integer
Iterations	14 to 100	Integer
Anchor ratio	10 to 30	Integer
Transmission range	12 to 25	Integer
ALE (in m)	0.39 to 2.57	Real

Table 2. Abbreviation of models and their settings.

Abbrev.	Model description
Quantum neurons	
QCN NM-O	Continuously neuron + Nesterov momentum
QCN GD-O	Continuously neuron + gradient descent
QCN A-O	Continuously neuron + Adam
QCN AG-O	Continuously neuron + Adagrad
QCN RMS-O	Continuously neuron + RMSProp
QCN SPSA-O	Continuously neuron + SPSA
QPN NM-O	Parametric neuron + Nesterov momentum
QPN GD-O	Parametric neuron + gradient descent
QPN A-O	Parametric neuron + Adam
QPN AG-O	Parametric neuron + Adagrad
QPN RMS-O	Parametric neuron + RMSProp
QPN SPSA-O	Parametric neuron + SPSA
Classical models	
SVR C100-G0.1	SVR (C=100, $\gamma = 0.1$, RBF kernel)
SVR C100-KL	SVR (C=100, $\gamma =$ auto, linear kernel)
SVR C100-C1-KP	SVR (C=100, coef0=1, $\gamma =$ auto, polynomial kernel)
MLPR	MLP (100), ReLU, max-iter=1000
MLPR 50-50	MLP (50,50), ReLU, max-iter=1000
MLPR 100-100	MLP (100,100), ReLU, max-iter=1000
LR	Linear regression
DTR	Decision tree (random state=0)
RDG	Ridge regression ($\alpha = 0.5$)
SGDR	Stochastic gradient descent regressor
KNNR	K-nearest neighbors regressor

4.2. Experimental Design 1

The database was divided into 70% for training and 30% for testing. The training data was used to adjust the model parameters. Once the models were trained, the test data was employed to evaluate their performance. This process was repeated 30 times to account for the statistical variability of the initial values in the quantum optimization models and the random division of the training and testing data.

In the experiments, each of the two quantum neurons had its parameters optimized using three optimizers: Nesterov momentum, gradient descent, and Adam. To compare the performance of these quantum models with existing classical models, we implemented six classical regression models: the multilayer perceptron regressor (MLPR), support vector regressor (SVR), decision tree regressor (DTR), linear regressor (LR), K-nearest neighbors regressor (KNNR), and stochastic gradient descent regressor (SGDR).

Tab. 2 details the parameters and abbreviations of the classical and QML models used in this work. The metric used as a fitness function to adjust the parameters of the quantum models was the root mean square error (RMSE), given by

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}, \tag{12}$$

where, for a data set with n elements, $\hat{y}_i$ is the i-th desired output and y_i is the i-th continuous output of the model. The classical models were also evaluated after training with the RMSE metric.

4.3. Dataset 2 (An indoor localization dataset)

The *UJIIndoorLoc* dataset focuses on wireless local area network fingerprint-based positioning technologies, commonly known as Wi-Fi fingerprinting [37]. This dataset contains fingerprint data collected from various devices positioned across three buildings at Jaume I University.

The database consists of 529 columns. The attributes ranging from 001 to 520 represent Wi-Fi signal-strength values that constitute a Wi-Fi fingerprint. Each signal from a wireless access point (WAP) is represented as a negative integer value in dBm, with -100 dBm indicating a very weak signal. In comparison, 0 dBm corresponds to a strong signal from the detected WAP. However, not all WAPs are detected in every scan. The signal levels for these undetected WAPs remain unchanged, using an artificial value of -100 dBm by default for WAPs that were not detected by the device [37]. The target attributes to be predicted, which include latitude, longitude, floor, and building ID, are located in the subsequent columns, providing crucial information about the devices' coordinates. Additional relevant information is available in the following columns, including specific spaces (e.g., offices, laboratories) and the relative position (inside/outside the space) where the capture was performed. Moreover, details about the user (userID), the Android device (PhoneID), and the timestamp of the Wi-Fi capture were not used in our experiments.

This dataset aims to train a model to predict coordinates from a given Wi-Fi fingerprint. Thus, by collecting fingerprints from only a few points in the environment and developing an efficient model for the problem, it is possible to create a complete map of the environment and accurately predict device coordinates.

4.4. Experimental design 2

The *UJIIndoorLoc* dataset is inherently high-dimensional, originally comprising more than 500 Wi-Fi signal strength features. To enable a fair and tractable comparison between quantum and classical learning models, a principal component analysis (PCA) preprocessing step was applied prior to training [38]. Importantly, this dimensionality reduction was performed uniformly for all evaluated models, including both quantum and classical regressors, ensuring that no model benefited from dataset-specific preprocessing. The reduction from 512 to four dimensions was performed. It is essential to note that both the quantum and classical models received input transformed by the PCA algorithm for this database, with no preprocessing specific to any model type.

The reduction from the original feature space to four principal components was motivated by practical and methodological considerations. From a quantum computing perspective, the complexity of the proposed quantum neuron models scales linearly with the input dimension. Consequently, reducing the number of input features is essential to make the execution of the quantum circuits feasible under current near-term quantum hardware constraints. Without such reduction, the resulting circuit width would exceed the capabilities of NISQ devices and realistic noisy simulators.

In the *UJIIndoorLoc* dataset, there are three variables to be predicted: latitude, longitude, and floor (or altitude). A discrete value represents the floor, while latitude and longitude are continuous. Together, these three values provide the coordinates needed to determine the devices' locations. Since quantum neurons have only one output, three models were created to predict each case.

In this database, the optimization strategy adopted for the quantum models was defined based on outcomes from the first experimental task, in which multiple optimizers were systematically evaluated. Specifically, the optimizer and hyperparameter configuration that achieved the best performance in predicting the ALE were retained. Consequently, the quantum models evaluated in this experiment were trained exclusively with the Adam optimizer using improved hyperparameter settings, and no additional optimizer comparisons were conducted at this stage. Both quantum and classical models were evaluated using the RMSE metric defined in Equation (12).

Table 3. RMSE values for the dataset 1 (WSNs), including the standard deviation (SD), the minimum, and the maximum errors for each regressor. The smallest, mean, and largest values are highlighted in bold for each type of regressor (classical and quantum).

Regressor	RMSE (±SD)	Minimal error	Maximum error
QPN - A-O	**0.080747** ± 0.009391	**0.074084**	**0.091488**
QPN RMS-O	0.085258 ± 0.011488	0.076591	0.098288
RDG	**0.095177** ± 0.018319	0.068345	0.148529
QPN - NM-O	0.095388 ± 0.008802	0.089878	0.105540
LR	0.095714 ± 0.016814	0.070698	0.145007
KNNR	0.095742 ± 0.017511	0.073327	0.150987
MLPR 100-100	0.099570 ± 0.022450	**0.065847**	0.162584
DTR	0.096537 ± 0.018023	0.068158	0.139696
SVR C100-G0.1	0.097704 ± 0.013898	0.072648	**0.130646**
SVR C100-KL	0.098460 ± 0.018574	0.073678	0.153337
SVR C100-C1-KP	0.103498 ± 0.015180	0.082130	0.139516
QPN AG-O	0.104683 ± 0.008236	0.096483	0.114848
QPN SPSA-O	0.111713 ± 0.008258	0.102346	0.117942
QPN - GD-O	0.149802 ± 0.006451	0.142367	0.153907
QCN SPSA-O	0.153444 ± 0.009672	0.142667	0.161372
QCN RMS-O	0.166018 ± 0.009515	0.157829	0.176456
QCN NM-O	0.166142 ± 0.009354	0.158355	0.176518
QCN A-O	0.166158 ± 0.009471	0.158073	0.176578
SGDR	0.173443 ± 0.027795	0.128509	0.240689
MLPR 50-50	0.216665 ± 0.017405	0.178731	0.249708
MLPR	0.228758 ± 0.024766	0.183519	0.297666
QCN AG-O	0.396798 ± 0.001016	0.395315	0.397726
QCN GD-O	0.481326 ± 0.002155	0.478852	0.482788

5. Results

5.1. Dataset 1

Tab. 3 presents the RMSE results obtained by classical and quantum regressors on Dataset 1 (WSNs). Among the classical approaches, Ridge Regression (RDG) achieved the best average performance, with an RMSE of 0.095177±0.018319, followed closely by LR, KNNR, DTR, and the SVR-based models, all with RMSE values around 0.096–0.103. These results indicate that several classical regressors were competitive on this task, particularly considering the diversity of inductive biases represented by linear, kernel-based, tree-based, and neural approaches.

Among the quantum models, the parametrized quantum neuron (QPN) variants consistently achieved the strongest results. In particular, the QPN optimized with Adam (QPN A-O) obtained the best overall performance across all evaluated models, reaching an RMSE of 0.080747±0.009391. This corresponds to an approximate relative reduction of 15% in average RMSE when compared with the best-performing classical regressor (RDG). The QPN trained with RMSProp (QPN RMS-O) also achieved competitive performance, with an RMSE of 0.085258, remaining below all classical baselines evaluated in this study.

The error distribution statistics further reinforce the consistency of the best QPN configurations. QPN A-O achieved the smallest maximum error among all models (0.091488), while also presenting a relatively low minimum error (0.074084). In contrast, although some classical models reached lower isolated minimum values in specific runs—such as MLPR 100-100 (0.065847) and RDG (0.068345)—their average RMSE and maximum-error values were substantially higher, indicating greater variability across executions. Therefore, the results suggest that the best-performing QPN models combined competitive average accuracy with more stable predictive behavior across repeated trials.

It is also important to note that the performance advantage was not uniform across all quantum configurations. While the QPN models optimized with Adam and RMSProp achieved highly competitive results, several QCN variants and optimization strategies produced significantly larger errors, particularly QCN AG-O and QCN GD-O. This variability indicates that the effectiveness of the quantum models depends strongly on both the neuron architecture and the optimization strategy adopted during training. Consequently, the results do not support a generalized claim that quantum regressors universally outperform classical methods, but rather that specific quantum parameterizations can achieve superior performance under suitable optimization conditions.

Given the strong performance of the QPN models optimized with Adam and RMSProp, additional experiments were conducted to refine their internal hyperparameters. The objective was to evaluate whether manually adjusted configurations could further improve predictive accuracy. However, no substantial gains were observed beyond the default settings. In particular, the default Adam configuration—stepsize=0.01, β_1=0.9, β_2=0.99, and $\varepsilon = 10^{-8}$ —already provided stable convergence and strong predictive performance for this regression task. These findings suggest that the optimization landscape of the QPN model can be effectively explored using standard adaptive optimization strategies, without requiring extensive hyperparameter engineering.

Overall, the results from Tab. 3 indicate that parametrized quantum neurons, particularly QPN A-O and QPN RMS-O, achieved the best predictive performance on Dataset 1, surpassing the evaluated classical baselines in terms of average RMSE while also exhibiting comparatively low worst-case errors. Although the advantage is concentrated in specific quantum configurations rather than across all quantum models, the results suggest that quantum-enhanced parameterizations may provide useful representational benefits for nonlinear regression problems in WSN environments.

Beyond the numerical comparison, the observed reduction in prediction error may have practical implications for WSN localization tasks. In this dataset, the best classical regressor achieved an RMSE close to 0.095, whereas the best-performing quantum model (QPN A-O) reduced this value to approximately 0.081. Although the absolute difference is relatively small, such reductions can still be relevant in localization-related applications, where accumulated estimation errors may propagate to routing, coverage estimation, and energy-management mechanisms.

Furthermore, the lower maximum error observed for QPN A-O suggests increased robustness across repeated executions. From an operational perspective, robustness is particularly important in WSN scenarios because large localization errors can disproportionately affect network performance and decision-making processes. Therefore, the improvements achieved by the best-performing quantum parametrized neurons should be interpreted not only as lower average prediction errors, but also as evidence of more stable predictive behavior under varying training conditions.

5.2. Dataset 2

Tab. 4 presents a comparative analysis between the best-performing quantum parametrized neuron configuration (QPN A-O) and a diverse set of classical regressors for the latitude, longitude, and floor prediction tasks of Dataset 2 (Wi-Fi fingerprinting). Only QPN A-O was selected as the quantum representative because previous experiments had already identified this configuration as the strongest quantum model, while the high computational cost associated with training quantum regressors made a broader evaluation of multiple quantum architectures impractical. The Wilcoxon signed–rank test was employed to assess whether the observed performance differences between QPN A-O and the classical regressors were statistically significant.

Table 4. Wilcoxon signed–rank test comparing the quantum parametrized neuron (QPN) with classical learning models across the Latitude, Longitude, and Floor prediction tasks for the Dataset 2 (Wi-Fi Fingerprinting). Only the QPN was evaluated on the quantum side due to its superior performance in earlier experiments and the high computational cost of training quantum models. Bold p-values indicate circumstances in which the quantum model cannot be considered statistically different from a classical counterpart (e.g., SGDR in the Floor task) or cases in which the QPN is statistically equivalent to several classical models (as observed in the Latitude and Longitude tasks).

	Latitude p-value	RMSE (±SD)	Longitude p-value	RMSE (±SD)	Floor p-value	RMSE (±SD)
QPN A-O	—	0.01561 ± 0.00056	—	0.00114 ± 0.0	—	0.30173 ± 0.00191
SVR C100-G0.1	**0.00391**	$0.01657 \pm$ 4e-05	0.00195	1e-05 ± 0.0	0.00195	0.27492 ± 0.0018
SVR C100-KL	**0.00391**	$0.01658 \pm$ 4e-05	0.00195	1e-05 ± 0.0	0.00391	0.30385 ± 0.00141
SVR C100-KP	**0.00391**	$0.01657 \pm$ 6e-05	0.00195	1e-05 ± 0.0	0.00195	0.27208 ± 0.00218
MLPR	**0.00391**	$0.01657 \pm$ 4e-05	0.00195	1e-05 ± 0.0	0.00195	0.25277 ± 0.00245
MLPR 50-50	**0.00391**	$0.01658 \pm$ 4e-05	0.00195	1e-05 ± 0.0	0.00195	0.21829 ± 0.00238
MLPR 100-100	**0.00391**	$0.01657 \pm$ 6e-05	0.00195	1e-05 ± 0.0	0.00195	0.20233 ± 0.00373
LR	**0.00391**	$0.01657 \pm$ 4e-05	0.00195	1e-05 ± 0.0	0.04883	0.30017 ± 0.0013
DTR	**0.00391**	$0.01658 \pm$ 4e-05	0.00195	1e-05 ± 0.0	0.00195	0.23239 ± 0.00292
RDG	**0.00391**	$0.01657 \pm$ 6e-05	0.00195	1e-05 ± 0.0	0.04883	0.30017 ± 0.0013
SGDR	0.00195	0.00526 ± 0.00042	**0.00195**	0.00438 ± 0.00036	**0.06445**	0.3002 ± 0.00129
KNNR	0.00195	0.00535 ± 0.00031	**0.00195**	0.00448 ± 0.00031	0.00195	0.19281 ± 0.00177

For the latitude prediction task, QPN A-O achieved an RMSE of 0.01561 ± 0.00056, demonstrating competitive predictive performance among the evaluated models. Although classical regressors such as SGDR and KNNR attained lower RMSE values, the quantum model numerically outperformed several other classical approaches, including the SVR-, LR-, RDG-, DTR-, and MLPR-based regressors, whose RMSE values remained close to 0.01657–0.01658. The Wilcoxon signed–rank test further indicates that the QPN remains statistically competitive with several classical methods, highlighting its ability to produce stable predictions for this task despite not achieving the lowest numerical error overall.

The *longitude* prediction task exhibited a different behavior. Although QPN A-O achieved a very small RMSE (0.00114), several classical regressors—particularly the SVR-based models—obtained substantially lower errors on the order of 10^{-5}. SGDR and KNNR also produced lower RMSE values than the QPN. Nevertheless, the Wilcoxon analysis revealed that QPN A-O remained statistically competitive with SGDR and KNNR, as reflected by the non-rejectable comparisons highlighted in bold. Therefore, despite the larger mean error observed for the quantum model, its predictive behavior could not be considered statistically distinct from these strong classical baselines. This result suggests that the QPN maintains competitive predictive consistency even in scenarios where it is not the numerically best-performing model.

For the *Floor* prediction task, QPN A-O achieved an RMSE of 0.30173 ± 0.00191. Although this result places the quantum model among the least accurate approaches for this task, it was not the worst-performing regressor overall. In particular, the SVR C100-KL model obtained a slightly larger RMSE (0.30385 ± 0.00141). Moreover, classical models such as SGDR, LR, and RDG achieved numerically similar errors, whereas models such as KNNR, DTR, and the deeper MLPR architectures produced substantially lower RMSE values. Even in this less favorable scenario for the quantum model, the Wilcoxon test indicated that QPN A-O could not be considered statistically different from SGDR ($p = 0.06445$). Thus, although the quantum model was numerically inferior to most classical regressors in the floor prediction task, its predictive error distribution remained statistically comparable to at least one strong classical baseline.

From an application perspective, the predictive performance achieved by QPN A-O in Dataset 2 may still have practical relevance for Wi-Fi fingerprinting-based indoor localization systems. Accurate coordinate estimation is essential for applications such as indoor navigation, asset tracking, context-aware services, and emergency response. In these scenarios, even moderate reductions in regression error may contribute to improvements in localization accuracy, particularly when prediction errors accumulate across multiple spatial dimensions. Furthermore, more stable coordinate estimation can reduce room or floor misclassification and improve the reliability of location-aware services in complex indoor environments.

Taken together, the Wilcoxon analyses demonstrate that the QPN exhibits robust and competitive performance across the three prediction tasks. It achieved the best overall behavior for the latitude task, remained statistically competitive with SGDR and KNNR in the longitude task despite presenting larger numerical errors, and could not be considered statistically different from SGDR in the floor prediction task. These findings reinforce the effectiveness of the QPN as a QML-based regressor and highlight an important characteristic of the quantum approach: even in scenarios where classical models achieve lower mean RMSE values, the predictive error distributions produced by the QPN may remain statistically comparable to those of strong classical baselines. Therefore, the results suggest that parametrized quantum neurons constitute a viable and competitive alternative for practical regression problems in Wi-Fi fingerprinting and related localization domains.

5.3. Computational Complexity and Scalability of Quantum Neuron Models

Beyond predictive error, computational complexity and scalability are fundamental aspects when assessing the practical applicability of learning models in wireless communication systems. In this regard, the quantum neuron architectures investigated in this work exhibit particularly favorable properties.

Both quantum models considered — the QCN and the QPN — are implemented as constant-depth variational quantum circuits whose width grows linearly with the number of input features. For an input vector of dimension m, the circuit requires m data-encoding operations and m corresponding parameterized quantum gates, followed by a fixed measurement stage on an ancillary qubit. As a consequence, the total number of quantum gates and trainable parameters scales as $\mathscr{O}(m)$.

From a training standpoint, each optimization step involves evaluating a single expectation value. Gradient-based optimizers rely on parameter-shift rules or equivalent gradient-estimation techniques, which introduce only a constant multiplicative overhead per trainable parameter. Therefore, the computational cost per training iteration also scales linearly with the input dimensionality. This behavior contrasts with that of classical models such as multilayer perceptrons or kernel-based regressors, whose training complexity typically grows superlinearly with the number of features and parameters.

This linear scaling is especially appealing in the context of near-term quantum hardware, commonly referred to as the noisy intermediate-scale quantum (NISQ) era [39]. In such environments, circuit depth, coherence time, and gate fidelity impose strict constraints on the execution of quantum workloads. The shallow depth and reduced parameter count of quantum neuron models mitigate noise accumulation, making them more suitable for execution on real quantum devices or realistic noisy simulators. Notably, the proposed models do not rely on deep entangling layers or wide multi-qubit structures, which are known to exacerbate training instability and hardware-induced errors.

Although the experiments reported in this work were conducted on noiseless quantum simulators, the architectural simplicity of the quantum neurons suggests that their computational footprint is well aligned with the limitations of practical quantum execution. In this sense, the reported performance gains should not be attributed to circuit depth or extensive quantum resources, but rather to compact quantum feature mappings implemented with minimal computational overhead.

Overall, these characteristics position quantum neuron-based regressors as lightweight quantum learning models whose computational complexity scales linearly with the input size. This property makes them promising candidates for future hybrid quantum-classical learning pipelines in wireless communication systems operating under NISQ constraints.

6. Conclusion

This study investigated the use of quantum neurons as regressors for predicting the mean localization error in wireless sensor networks and for estimating the positions of mobile devices using Wi-Fi fingerprint-based localization. Two quantum neuron architectures were trained using multiple optimization strategies and compared against a diverse set of classical regressors, including linear models, support vector machines, multilayer perceptrons, decision trees, k-nearest neighbors, and stochastic gradient descent. The main objective was to assess whether small-scale quantum circuits equipped with parameterized gates can effectively address real-world regression tasks involving nonlinear and heterogeneous data.

Across both evaluated datasets, the results demonstrated that quantum neurons—particularly the parametric neuron trained with the Adam optimizer—achieved competitive predictive accuracy. In Dataset 1, the quantum model not only outperformed all classical regressors in terms of root mean squared error but also exhibited more stable minimum and maximum error bounds, highlighting its expressive capacity in complex regression scenarios. In Dataset 2, the quantum model achieved the best performance for latitude prediction, was statistically indistinguishable from strong classical competitors (SGDR and SVR) in longitude prediction, and remained competitive in floor prediction, where it was statistically equivalent to SGDR despite presenting a slightly higher numerical error. These findings indicate that quantum neural circuits can operate at a level comparable to—and in some cases superior to—well-established classical methods.

The experiments also provided important insights regarding training and model design. First, the default hyperparameters of the Adam optimizer were found to be nearly optimal for training quantum neurons, suggesting that effective learning can be achieved without extensive hyperparameter tuning. Second, the study demonstrated that shallow, low-width quantum circuits can represent complex regression functions, reinforcing the feasibility of quantum learning under hardware constraints typical of the NISQ era. Third, Wilcoxon statistical analyses revealed several scenarios in which quantum and classical models exhibit statistically equivalent performance, a particularly relevant observation given that quantum models often rely on significantly fewer trainable parameters.

Overall, this work advances quantum machine learning by demonstrating that quantum neurons constitute a viable and competitive alternative for nonlinear regression tasks. The empirical evidence suggests that small-scale quantum circuits can capture rich functional dependencies, offering advantages in expressivity and statistical behavior. As quantum hardware continues to evolve, these results motivate further investigations into quantum neural architectures, their optimization landscapes, and their potential to complement or surpass classical approaches in practical machine learning applications.

Although this study focused on compact quantum neuron architectures to evaluate their expressivity and feasibility under strict NISQ constraints, an important direction for future research is the systematic comparison with more expressive variational quantum circuit architectures. This includes, among others, deeper quantum models based on structured entanglement layers, data re-uploading strategies, and hardware-efficient ansätze. A detailed analysis of these architectures will enable a more comprehensive evaluation of the trade-offs between expressivity, trainability, circuit depth, and computational cost. Furthermore, extending the current framework to alternative variational quantum models may help clarify the relative advantages of quantum-neuron-based regressors and identify scenarios in which more complex quantum architectures offer additional benefits for localization and communication-related learning tasks.

Acknowledgements

Fernando M. de Paula Neto acknowledges financial support from the Brazilian National Council for Scientific and Technological Development (CNPq) under Grant No. 408499/2025-7. Fernando M. de Paula Neto is also a researcher affiliated with the National Institute of Science and Technology for Applied Quantum Computing, supported by CNPq under Grant No. 408884/2024-0, and is part of the Brazilian National Institute for Artificial Intelligence, supported by CNPq under Grant No. 406417/2022-9.

Conflicts of Interest

The authors declare no competing interests.

References

[1] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM Journal on Computing 26 (5) (1997) 1484–1509.

[2] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, S. Lloyd, Quantum machine learning, Nature 549 (7671) (2017) 195–202.

[3] H. Zhao, A. Zlokapa, H. Neven, R. Babbush, J. Preskill, J. R. McClean, H.-Y. Huang, Exponential quantum advantage in processing massive classical data, arXiv preprint arXiv:2604.07639 (2026).

[4] R. Rahman, D. C. Nguyen, Quantum convolutional neural networks: A survey on architectures, applications, and future directions, IEEE Transactions on Neural Networks and Learning Systems (2026).

[5] A. Verdone, F. Succetti, A. Ceschini, A. Rosato, A. Fioravanti, M. Panella, A hybrid quantum-neural network for heart disease classification, Biomedical Signal Processing and Control 113 (2026) 109185.

[6] S. Mangini, F. Tacchino, D. Gerace, C. Macchiavello, D. Bajoni, Quantum computing model of an artificial neuron with continuously valued input data, Machine Learning: Science and Technology 1 (4) (2020) 045008.

[7] J. H. A. de Carvalho, F. M. de Paula Neto, Parametrized constant-depth quantum neuron, IEEE Trans. Neural Netw. Learn. Syst. 35 (11) (2024) 15932–15943. doi:10.1109/TNNLS.2023.3290535.

[8] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, S. Woerner, The power of quantum neural networks, Nature Computational Science 1 (2021) 403–409.

[9] J. Jäger, R. V. Krems, Universal expressiveness of variational quantum classifiers and quantum kernels for support vector machines, Nature Communications 14 (1) (2023) 576.

[10] M. Schuld, R. Sweke, J. J. Meyer, Effect of data encoding on the expressive power of variational quantum-machine-learning models, Physical Review A 103 (3) (2021) 032430.

[11] S. Chen, S. Yoo, Y. Fang, et al., Universal approximation theorem and error bounds for quantum neural networks and quantum reservoirs, IEEE Transactions on Neural Networks and Learning SystemsEarly access / preprint versions may exist (2022).

[12] F. M. Neto, Data complexity measures for quantum circuits architecture recommendation, arXiv preprint arXiv:2502.15129 (2025).

[13] F. M. Neto, L. d. R. Silva, P. S. Neto, F. F. Fanchini, Regression of functions by quantum neural networks circuits, arXiv preprint arXiv:2512.19978 (2025).

[14] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, H. Neven, Barren plateaus in quantum neural network training landscapes, Nature communications 9 (1) (2018) 4812.

[15] R. D. Timoteo, D. C. Cunha, A scalable fingerprint-based angle-of-arrival machine learning approach for cellular mobile radio localization, Computer Communications 157 (2020) 92–101.

[16] H. Ge, Z. Sun, Y. Chiba, N. Koshizuka, Accurate indoor location awareness based on machine learning of environmental sensing data, Computers & Electrical Engineering 98 (2022) 107676.

[17] S. Kagi, B. S. Mathapati, Localization in wireless sensor network using machine learning optimal trained deep neural network by parametric analysis, Measurement: Sensors 24 (2022) 100427.

[18] H. Attar, W. Saber Ismail, M. Hafez, S. Bahaa, M. A. Deif, M. Khosravi, H. Youssry, Machine learning in indoor localization prediction using received signal strength indicator and Wi-Fi network, Int. J. Intell. Networks 6 (2025) 233–243.

[19] A. Gopakumar, L. Jacob, Localization in wireless sensor networks using particle swarm optimization, in: 2008 IET International Conference on Wireless, Mobile and Multimedia Networks, IET, 2008, pp. 227–230.

[20] S. Goyal, M. S. Patterh, Wireless sensor network localization based on Cuckoo search algorithm, Wireless Personal Communications 79 (1) (2014) 223–234.

[21] J. Cheng, L. Xia, An effective cuckoo search algorithm for node localization in wireless sensor network, Sensors 16 (9) (2016) 1390.

[22] A. Singh, V. Kotiyal, S. Sharma, J. Nagar, C.-C. Lee, A machine learning approach to predict the average localization error with applications to wireless sensor networks, IEEE Access 8 (2020) 208253–208263.

[23] F. Meneses, A. Moreira, A. Costa, M. J. Nicolau, Radio maps for fingerprinting in indoor positioning, in: J. Conesa, A. Pérez-Navarro, J. Torres-Sospedra, R. Montoliu (Eds.), Geographical and Fingerprinting Data to Create Systems for Indoor Positioning and Indoor/Outdoor Navigation, Academic Press, 2019, pp. 69–95. doi:https://doi.org/10.1016/B978-0-12-813189-3.00004-6.

[24] Y. Wang, R. Sun, Q. Cheng, W. Y. Ochieng, Measurement quality control aided multisensor system for improved vehicle navigation in urban areas, IEEE Trans. Ind. Electron. 71 (6) (2024) 6407–6417. doi:10.1109/TIE.2023.3288188.

[25] T. Koike-Akino, P. Wang, Y. Wang, AutoQML: Automated quantum machine learning for Wi-Fi integrated sensing and communications (2022). arXiv:2205.09115. URL https://arxiv.org/abs/2205.09115

[26] S. Ahmed, M. Youssef, Device-independent quantum fingerprinting for large scale localization, in: 2022 20th Mediterranean Communication and Computer Networking Conference (MedComNet), IEEE, 2022, pp. 208–215.

[27] S. Ahmed, M. Youssef, Quantum fingerprinting for heterogeneous devices localization, Computer Communications 204 (2023) 43–51.

[28] E. Guler, M. T. Kakız, T. Çavdar, F. B. Günay, B. Şanal, A quantum machine learning approach for detecting user locations, Journal of Millimeterwave Communication, Optimization and Modelling 3 (1) (2023) 1–8.

[29] M. Jeong, H. Park, S. W. Choi, S. Kim, et al., Fingerprinting-based indoor localization with hybrid quantum-deep neural network, IEEE Access 11 (2023) 142276–142291.

[30] S. J. Nawaz, S. K. Sharma, S. Wyne, M. N. Patwary, M. Asaduzzaman, Quantum machine learning for 6g communication networks: State-of-the-art and vision for the future, IEEE Access 7 (2019) 46317–46350.

[31] M. A. Nielsen, I. L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2000.

[32] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice Hall PTR, 1994.

[33] M. Benedetti, E. Lloyd, S. Sack, M. Fiorentini, Parameterized quantum circuits as machine learning models, Quantum Science and Technology 4 (4) (2019) 043001.

[34] Y. Huang, H. Lei, X. Li, An empirical study of optimizers for quantum machine learning, in: 2020 IEEE 6th International Conference on Computer and Communications (ICCC), IEEE, 2020, pp. 1560–1566.

[35] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, et al., Variational quantum algorithms, Nature Reviews Physics 3 (9) (2021) 625–644.

[36] V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, S. Ahmed, V. Ajith, M. S. Alam, G. Alonso-Linaje, B. AkashNarayanan, A. Asadi, et al., Pennylane: Automatic differentiation of hybrid quantum-classical computations, arXiv preprint arXiv:1811.04968 (2018).

[37] J. Torres-Sospedra, R. Montoliu, A. Martínez-Usó, J. P. Avariento, T. J. Arnau, M. Benedito-Bordonau, J. Huerta, UJIIndoorLoc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems, in: 2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN), IEEE, 2014, pp. 261–270.

[38] R. Bro, A. K. Smilde, Principal component analysis, Analytical Methods 6 (9) (2014) 2812–2831.

[39] J. Preskill, Quantum computing in the nisq era and beyond, Quantum 2 (2018) 79.