Modified Deep Residual Quantum Computing Optimization Technique for IoT Platform

Rasha M. Abd El-Aziz, Alanazi Rayan, Osama R. Shahin, Ahmed Elhadad, Amr Abozeid, Ahmed I. Taloba
Department of Computer Science, College of Science and Arts in Qurayyat
Jouf University, Saudi Arabia

***Abstract*—Internet of Things (IoT) is defined as millions of interconnections between wireless devices to obtain data globally. The multiple data are targeting to observe the data through a common platform, and then it becomes essential to investigate accuracy for realizing the best IoT platform. To address the growing demand for time-sensitive data analysis and real-time decision-making, accuracy in IoT data collecting has become critical. The Res-HQCNN is a hybrid quantum-classical neural network with deep residual learning. The model is qualified in an end-to-end analog method in a traditional neural network, backpropagation is used. To discover the Res-HQCNN efficiency to perform on the classical computer, there has been a lot of investigation into quantum data with or without noise. Then focus on the application of the artificial neural network to analyze the dangers to these IoT networks. For data recording purposes, to undertake in-depth analysis on the threat severity, kind, and source, a model is trained using recurrent and convolutional neural networks. The intrusion detection system (IDS) explored in this study has a success rate of 99% based on the empirical data supplied to the model. Due to irregularly distributed robust execution, larger affectability for the introduction of authority dimension, steadiness, and the extremely large crucial area, a quantum hash function work has been proposed as an amazing method for secure communication between the IoT and cloud.**

***Keywords*—Internet of things (IoT); cloud; Res-HQCNN; intrusion detection system (IDS); optimization**

I. Introduction

Artificial neural networks (ANN) are one of the most successful computational approaches. Neural network-based machine learning algorithms are improving and advancing [1]. In the machine learning sector, neural networks are currently enjoying remarkable success and have a wide range of applications, including pattern recognition, video analysis, medical diagnosis, and robot control. Quantum neural networks (QNN) appear in parallel with the development of artificial neural networks (ANNs), with the promise of overcoming classical computation limits using quantum computing [2]. The paper shows a quantum feed - forward neural network made up of genuinely quantum neurons. It has a remarkable capacity to study an unknown homogeneous and a high level of robustness when dealing with noisy training data. Due to a decrease in the number of coherent qubits, this process is essential for noisy approximate quantum computers. The adherence among a pure quantum system and an arbitrary quantum state is selected also as cost function in this study. However, as the number of network layers grows larger, the convergence rate of a cost function slows, and the value of convergence even fails to deliver the highest for clean data is shown in Fig. 1(a). In the case of noisy data, Fig. 1(b) shows it as the system gets deeper, the strength for noisy data weakens. As a result, guess if the cost function's efficiency can be enhanced both for clean and noisy data. To show the number of convolution layers in the corresponding layer, use a one-dimensional list of real numbers.

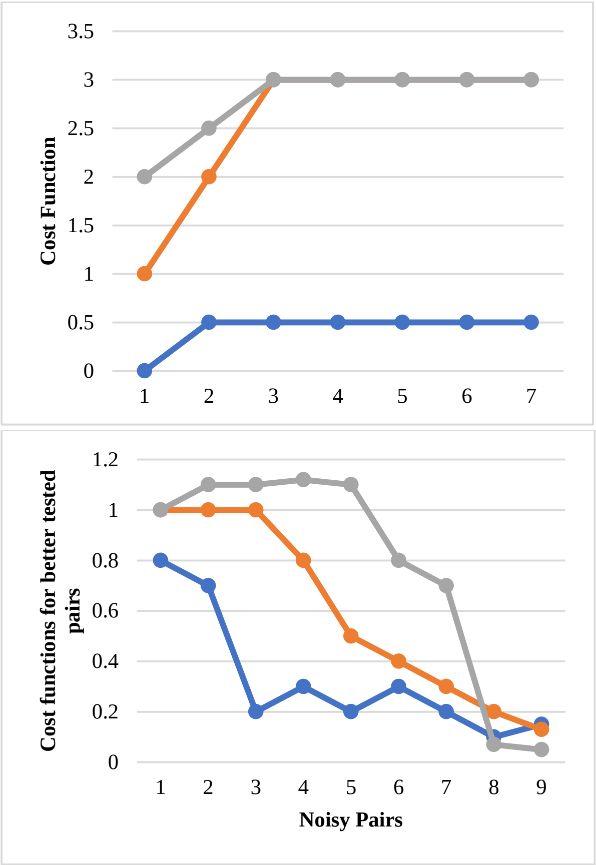


Fig. 1. QNN Numerical Results for Clean and Noisy Data.

To enhance the novel quantum-classical neural network with the deep residual learning (Res-HQCNN) to attain a goal, inspired by the deep residual efficiency learning. This is a novel concept and no work has been attempted as far as know. To find the efficiency incorporate a residual scheme into QNNs. It is not a simple task [3]. The amount of residual block structures, levels of the network count, and whether or not to skip the layer all have an impact on the parameter updating method. Because the informing parameters of the matrix are derived from the function of derivative formulation, the updating parameters matrix for each network is unique. Res-HQCNN now outperforms previous QNNs on both fresh and loud quantum data while requiring only a conventional machine to build. Discuss a different way for incorporating quantum neural networks with residual block structure so that they can be executed on quantum computers.

Networks intrusion continues to occur. This is even though numerous artificial intelligence techniques have been created throughout time to prevent such incidents [4]. Various advancements and modifications are made to network configuration protocols daily to improve them, but in reality, there is a weakening risk of the protocols with or without understanding. Malware and other intrusions frequently take advantage of minute alterations made to the original core development codes that serve as the foundation for running and maintaining networks. The changes are vital, but they come at a high price. It’s time to rethink your threat management strategy [5]. IoT and Cloud Computing benefit in the same way and Cloud Computing is constantly encouraged to improve the introduction to the level of high resource usage, accumulation, necessity, and processing capability.

However, network intrusions continue to occur. This is regardless of the fact that multiple artificial intelligence techniques have been developed over time to prevent such incidents. Various progressions and modifications are made to configuration management protocols on a daily basis in an effort to improve them, but in reality, there is a risk of decimating the procedures with or without knowledge [6]. Malware and other intrusions frequently take benefit of minute changes respect to the design foundational development rules that serve as the foundation for operating and maintaining networks. The changes are needed, but they come at a high cost.

Cloud is a ground-breaking platform that can provide additional features as an information distribution delegate. When an IoT client has valid requests for specific information to be acquired, stored, and accessed, he can simply designate the requests to the cloud whenever with more remarkable comfort. A couple of incites linked to contraption disillusionment are addressed by cloud and IoT applications developed in resource-constrained conditions. A QHF is proposed to address IoT security concerns. It converts an old-style message to a Hilbert space, preventing programmers from obtaining too much information about the old-style message. Safety issues are of extreme importance, and they must be addressed without exacerbating the system's or devices' dimensions [7]. A few calculations concerning safety issues have been published in prior studies. The U-2 hash work is the largest class of hash capacity groups among known hash capacity groups, assuring good safety.

The number of residual block structures, the number of network layers, and whether or not to skip layers all have an impact on the variable propagation algorithm. Because the updating parameters structure is derived from the description of the derivative feature, the updating parameters structure for each network structure is unique. The updating parameters matrix becomes more complex as the network structure becomes more varied. As a result, this investigation is both challenging and intriguing. This hope that our paper will serve as a useful resource in this field of study. The following are some of the contributions made as a result of this paper:

- Develop a new residual learning structure that is focused on QNNs.
- Calculate the current training algorithm using the Res-HQCNN model. Examine the performance from the level of information propagation feedforward and backward, subset of the training algorithm.
- concentrates on using Artificial Neural Networks to evaluate the risks to such IoT networks. For data acquisition reasons, a classifier is constructed using recurrent and convolutional neural networks to perform effective analysis on threat intensity, type, and source.
- Res-HQCNN has better performance across both clean and noisy quantum information than previous QNNs at the cost of implementation.

The remaining part and the aim of this paper have explained the RES-HQCNN optimization technique for IoT; Section 2 defines the highlight of the previous effort that can be done by the scholars in this domain; Section 3 offering the methodology architecture model and its mechanism, Section 4 represents the result and discussion and Section 5 represents the work achieved in conclusion and future work.

II. Related Work

The author in [8] evaluates Quantum Computing (QC) has grown in popularity as a result of its unique characteristics, which, in terms of performance and operation methods, differ from typical computers This research proposes hybrid models and approaches for large-scale mixed-integer programming issues that successfully combine the complementary strengths of deterministic algorithms and quality control techniques to solve a combinatory difficulty. Large-scale instances of these application problems across multiple dimensions, ranging from molecular design to logistics optimization, are computationally demanding for deterministic optimization algorithms on classical computers. To address the computing challenges, hybrid QC-based approaches are suggested, with comprehensive computational experimental results demonstrating their pertinence and productivity. The suggested QC-based solution approaches offer high computational efficiency in terms of solution quality and computation time by leveraging the unique properties of both classical and quantum computers.

The author in [9] introduces a Deep residual network with adequate depth but bounded width has recently been proven to be capable of universal approximation in the sense of the supremum norm. Illustrate to adapt existing deep residual network training methods to establish approximation bounds for the test error in the supremum norm based on the training error using these results. This technique is based on control-theoretic interpretations of these networks in discrete and continuous time, and they show that constraining the set of parameters to be learned in a way that is consistent with most commonly used training procedures is sufficient.

The author in [10] is proposed to use a combination of modified deep learning and reinforcement learning in an incentive-based demand response (DR) algorithm. A modified deep learning model based on recurrent neural network (MDL-RNN) was initially suggested to forecast future environmental uncertainties by projecting day-ahead wholesale energy price, photovoltaic (PV) power output, and power load. Then, using reinforcement learning (RL), researchers looked at the best incentive rates for each hour that would maximize earnings for both ESPs and EUs. When compared to other methods, the findings demonstrated that the proposed upgraded deep learning model can produce more precise forecasting predictions A short-term DR program was developed for peak electricity demand periods, and trial results show that peak electricity demand can be reduced by 17%. This helps to improve power system security by reducing supply-demand imbalances.

The author in [11] improve the feature mapping process, introduce a hybrid quantum-classical convolutional neural network (QCCNN), which is based on convolutional neural networks (CNNs) but is optimized for quantum computing. In terms of both the number of qubits and the depths of the circuits, QCCNN is favorable to existing noisy intermediate-scale quantum computers, while keeping crucial aspects of classical CNN, such as nonlinearity and scalability. Also, offer a methodology for computing the gradients of hybrid quantum-classical loss functions automatically, which may be extended to other hybrid quantum-classical algorithms directly. By using a Tetris dataset to demonstrate the architecture's capabilities, show that QCCNN can perform classification tasks with learning accuracy that exceeds that of standard CNN.

The author in [12] analyze in classical systems, Control parameter optimization is frequently achieved using supervised machine learning and reinforcement learning; however, in quantum systems, parameter optimization is primarily accomplished using gradient-based greedy methods. To use differential evolution methods to avoid the non-convex optimization stagnation problem. To improve quantum control fidelity for noisy systems by averaging across the objective function. To reduce processing costs, this paper proposes methods for early run termination and adaptive search subspace selection. The implementation is massively parallel and vectorized to further reduce execution time. Quantum phase estimation and quantum gate design are two instances where these methods outperform greedy algorithms in terms of fidelity and scalability.

III. Proposed Methodology

In this section, the Res-HQCNN architecture model is defined based on QNN. According to the mechanism Res-HQCNN is defined based on a training algorithm.

A. *Architecture Model of Res-HQCNN*

In Res-HQCNN describe a residual block structure. The Res-HQCNN structure with many layers is offered. Thus offer Res-HQCNN examples with a unit hidden layer to further understand the mechanism [1]. Finally, examine the difference between the past QNNs and the Res-HQCNN. Thus, Fig. 2 shows the residual block diagram. In Res-QCNNN, a new residual block diagram is defined as follows by including a few assumptions and notations at the start for your convenience.

The procedure for combining the residual block structure with the quantum neural network in Res-HQCNN layer k defines a quantum perceptron as an arbitrary unitary operator with U_{k-1} input qubits and one output qubit. For L=1,2,…U_k the quantum perceptron $Q^k{}_L$ is a (U_{k-1}+1) for qubit unit. Quantum perceptron's with K hidden layers make up the Res-HQCNN. It uses the layer unitary operator Q^k in the form of a matric product of quantum perceptron's to work on an input state $\rho^{k_{in}}$ of input qubits and obtain a mixed state k+1 out for the output qubits: $Q^k = Q^k_{u1} Q^k_{u1-1} \ldots . Q^k_1$. For 1,2,…$Q_{k,}$ acts on the qubits in layers k-1 and 1, because the unitary operators are arbitrary and do not always commute, the layer unitary order is critical. The residual block structure provides the new input state for layer k+1 by adding the input state with the output state of layer k for k=1,2, …. During the processing of information from 1 into K+1 out and K.

In Fig. 3, "Res" denotes the Res-HQCNN residual block structure. The "Res" can be connected not just layer by layer continuously, but also by skipping one or more levels [13]. Res-architecture HQXNN propagates data from input to output, progressively passing through a network of quantum feeder neurons.

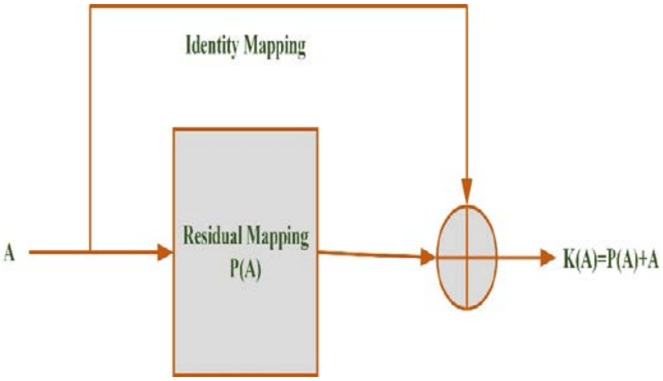


Fig. 2. Residual Block Diagram.

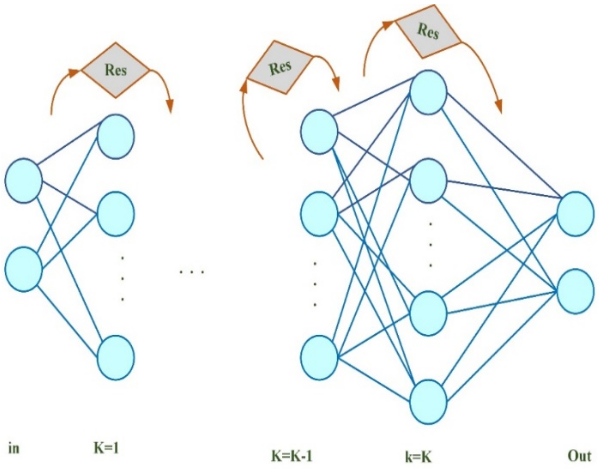


Fig. 3. Res-HQCNN Architecture.

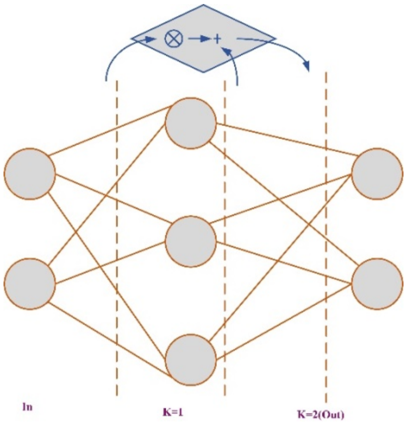


Fig. 4. Res-HQCNN Architecture with unit Hidden Layer.

The mechanism example for Res-HQCNN with one hidden layer in Fig. 4 helps with the comprehension. $Q^1 = Q_3^1 Q_2^1 Q_1^1$, which is a matrix product of quantum perceptron. The layer unitary between the input layer and the hidden layer is defined as $Q^2 = Q_2^2 Q_1^2$. The quantum perceptron's are applied layer by layer from top to bottom in the first stage, and the output state ρ^{1out} of the hidden layer is then computed as;

$$\rho^{1out} = is_{in}(Q^1(\rho^{1in} \otimes |000\rangle_{hid}\langle 000|)Q^{1^+})$$

Then, apply the residual block diagram to ρ^{1in} and ρ^{1out} to get a new input state for the output layer:

$$\rho^{2in} = \ \rho^{1out} + (\rho^{1in} \otimes |0\rangle\langle 0|)$$

In the following step, to obtain the final output state of Res-HQCNN from Fig. 5:

$$\rho^{1out} = is_{hid}(Q^2(\rho^{2in} \otimes |00\rangle_{out}\langle 00|Q^{2^+})$$

When comparing the previous QNNs to Res-HQCNN, that notice the trace value of the input state ρ^{k+1in} for some k changes as a result of the addition operation in the residual block structure [14]. Indicate k=2, $\rho^{2in} = \left(\rho^{1in} \otimes \left|0\right\rangle_{u_1-u_0}\left\langle 0\right|\right) + \rho^{1out}$, and $\rho^{3in} = \left(\rho^{2in} \otimes \left|0\right\rangle_{u_2-u_1}\left\langle 0\right|\right) + \ \rho^{2out}$, next trace values of the ρ^{2in} and ρ^{3in} are the 2 and 4, respectively. In theory, ρ^{2in} and ρ^{3in} are not density matrices, hence the training procedure cannot be used in a quantum computer. Every coin, however, has two sides. The residual block structure increases the cost function's performance, especially for deeper networks, as shown in the experiment section. It's also worth noting that the residual block structure can be applied to all concealed layers except the last output layer. Assumed $U_{k-1} \leq U_k$ for k = 1,2,….k and $U_0 = U_{k+1,}$ then the qubits in layer k in general. The final output of the network will be $\rho^{out} = \ \rho^{k+1in} + \rho^{k+1out}$ if the residual block structure is applied to the ρ^{k+1out} because the dimension of the ρ^{k+1in} is the equal to the dimension, that should use the partial trace on the ρ^{k+1in} to maintain the matrix addition rule.

The previously stated, the Res-HQCNN residual block structure has trouble similar to individuality mapping. But, by doing it will lose some ρ^{k+1in} information, which is incompatible to adopt the residual technique [15]. This also highlights the inefficiencies of applying residual block structure to the last output layer via an experiment.

B. *Res-HQCNN Training Algorithm*

N number pairs of training statistics, that are possibly unknown by the quantum states, are randomly generalized in the form of $(|\emptyset_a^{in}\rangle, |\emptyset_a^{out}\rangle)$ with a=1, 2,….., and N. It is also permissible to employ adequate copies of a training pair $(|\emptyset_a^{in}\rangle, |\emptyset_a^{out}\rangle)$ of a given a to avoid quantum projection noise, when compared to the cost functions derivative [16]. The intended output $|\emptyset_a^{out}\rangle$ as $|\emptyset_a^{out}\rangle$ = T $|\emptyset_a^{in}\rangle$ is choose to consider with an T as unknown unitary operation.

The cost function is used based on the mean fidelity of the Res output HQCNN and the expected results for all training data. However, to define the Res-HQCNN cost function to divide 2v, where v is the residual block number structures in Res-HQCNN, according to the residual block definition structure and fidelity linear fidelity:

$$R(f) = \frac{1}{2^v N} \sum_{a=1}^{N} (\,\emptyset_a^{out} | \rho_a^{out}(f) | \emptyset_a^{out}\rangle$$

To know the near network output state and the desired output state are, the higher fidelity. If the cost function equals 1 and 0, consider the Res-HQCNN to be the best performer to be the worst. As a result, the goal in the training process is to maximize the cost function. For each Res-HQCNN layer, that denote ρ_a^{lin} as the layer input state l and ρ_a^{lout} as the output layer state l with l=1,2,….L and a = 1,2,…….N. Consider the scenario in which each layer is added with a residual block structure and there is no skipping layer, then v=L. The Res-HQCNN training algorithm is explained in the following flow chart in Fig. 5.

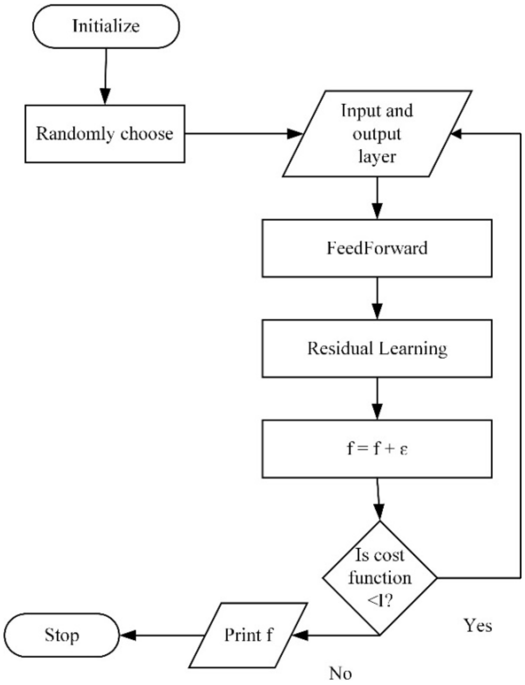


Fig. 5. Training Algorithm Flowchart.

Using a quantum hash function secure the data communication with encryption and decryption. They may occur some threat intrusion. Some types of threats and intrusion are following.

C. Types of Threats

By encrypting and decrypting data using the quantum hash function, data communication can be secured. They may be subjected to some sort of danger incursion [17]. Threats and intrusions of all kinds are on the way. Threats come in various forms:

- Malware.
- Data Loopholes.
- Feeble IoT network outlines.
- Service Denial.

D. Types of Intruders

The types of intruders are trying to intoxicate the network are following:

1) Outer intruder: This is an intruder from a different network from the one they're attempting to intoxicate. They use other networks, but they come to the network to distribute threats and recover data, among other things.

2) Inner network: This is an invader from a network other than the one they're trying to infect. They use other networks, but they come to the network for a variety of reasons, including propagating threats and recover data.

The internet connectivity of intruders on both online and offline:

1) Online Intruder: A danger has been identified as emanating from an internet source. This is particularly prevalent because they take advantage of relatively common IP addresses and can simply steal information from users of the addresses by messing with the network's coded backdrop.

2) Offline intruder: This is an invader who has gained access to the network but does not have internet access. There is virtually little technology available to deal with and counter this type of intrusion threat, yet this is a highly dangerous group of people.

E. Threat Proximity

This information is necessary to demonstrate the degree to which a network user is close to the threat described in the studies [18]. Unfortunately, this method can only be accessed by users of the same network. Due to differences in the functioning of the network, it would be more difficult to draw such a conclusion in the case of an external incursion.

To carry out the threat analysis, the input threat is exposed to a combination architecture of RNN and CNN that chucks the data into bits [19]. The data has a gaussian relationship, and it is assumed that the eventual output, after categorization and regrouping, will be a Gaussian distribution in a very precise manner. The following algorithms were used on the training model:

- Levenberg-Marquardt Algorithm: This approach has been utilized for neural network optimization and is highly useful because the threat is measured on a summation basis [20]. The intrusion is described as a collection of minor threats that add up to a level that is regarded as a threat numerically. Because desire a predefined category of various clusters, the neural network was trained with an input that specifies a certain objective.
- Feed Forward Algorithm: The connections employed in the node do not establish a rotating back dependence, which is ideal for the study. This algorithm is used to train the nonlinear optimization model [21]. It is represented in mathematical as:

  $$f(a) = \frac{1}{1+e^{-a}}$$

  $$f'(a) = f(a)\big(1 - f(a)\big)$$
- Backward learning algorithm: Sensitivity to the impacts of the feed-forward approach for model training. As a result, the feed-forward is primarily reliant on derivative functions, resulting in anticipation. Backward training is a strategy for optimizing a model that involves integration techniques [22]. It minimizes J and so optimizes the cost function for the Jacobian Matrix application.

F. Quantum Hash Function

The hash capacity is presented just in one-way great detail. Selecting the work verification, all single-direction QW work is considered [23]. The single path, solid impact opposition, and fragile crash obstruction are the main characteristics of H-work. The following are the quantum hash attributes capacities:

1) One-direction: It is possible to process the S regard S(G) by giving a data G, but it is computationally impossible to discover the basic data G with a given S regard S(G).

2) Frail crash obstruction: Based on the G data, it is impossible to find another data by computer G1 so that S(G)=S(G1).

3) Solid effect opposition: It is computationally impossible to locate the optional two unmistakably data G and G1 such that S(G)=S(G1). When grasping an S work, these three qualities are key models to consider. Quantum hash work, in comparison to old-style hash work, has more favorable circumstances, such as simple execution and a higher degree of security. Our information verification strategy will become more secure over time. the quantum hash work's nitty-gritty technique is depicted in the diagram below.

Parameter to be selected are [c, $\theta 1, \theta 2, \tau$] under the requirements: c is an odd number and {0< $\theta 1, \theta 2, \tau < \frac{\pi}{2}$} here τ – coin state$|0\rangle = \cos\tau|0\rangle + \sin\tau|1\rangle$, c- number of cycles. In addition, $\theta 1$ and $\theta 2$ are the two controllers of C-QW. The two-coin admin controllers are $\varphi^1 and\ \varphi^2$.

$$\varphi^1 = \begin{bmatrix} \cos\theta 1 & \sin\theta 1 \\ \sin\theta 1 & -\cos\theta 1 \end{bmatrix} \quad \varphi^2 = \begin{bmatrix} \cos\theta 2 & \sin\theta 2 \\ \sin\theta 2 & -\cos\theta 2 \end{bmatrix}$$

The underlying one-information bit selects "0" as its value of φ^1 chooses φ^2. The likelihood dispersion is created by rolling one coin and walking DTQW on a cycle substantially influenced by information G [24]. To frame a twofold H computation, multiply all qualities in the following likelihood circulation by 10i times and maintain only their entire number component modulo 2j with a ≥ b. The S respect has a bit length of mj. This is the methodology used in the most recent QH works conspire, which has a higher level of safety than previous ones. To deservedly chose this QH capability as the approval work.

G. Encryption

The encryption framework works with given data and is a key to creating a figured data that may be delivered through insecure channels without risk of being deciphered by anyone that doesn’t have the interpreting key. The key was initially subject to two sets of keys, one open and one private, for security concerns [25]. Initially, to encode, then to untangle, and finally in a different way; this is achievable due to the usage of particular mathematical constraints, which have non-reversible features.

Encryption = S (amount of A/T, S, \open key, A/T)

Decryption = ((m^k) | modified A/T (W)\) * open key

Hash efficiently communicates on little information to produce a string with a known length of G. The IoT sight and sound data are validated by the quantum value, open keys, and restricted irregularity has been able to abuse reduced the S-esteem in light of this worth [26]. The quantum value, open keys, and restricted irregularity have been able to exploit bargain the H-esteem in light of this worth to certify IoT sight and sound data.

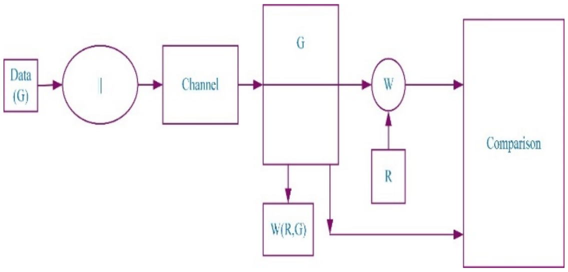


Fig. 6. Hashing Mechanism for Data Authentication.

Fig. 6 demonstrates the general information procedure confirmation. G is the data that will be transferred from the sender to the recipient. W is denoted as verification work is used to scramble the primary information of G. “| |” is a technique for teaching the underlying knowledge as well as the figure script. During correspondence, the square edge is used to symbolize the channel. The key that is utilized to encrypt the underlying data is R.

IV. Result and Discussion

The result examines the Res-HQCNN robustness to noisy quantum data. To compare to employ the same rule to test the robustness. The numerical output from running the two neural networks, RNN and CNN, is shown in the results. As can be seen, the entire input is fed into two robust neural platforms, which optimize the model that has been trained using learning algorithms, and the output is a classification of threats with subfolders indicating the severity of the threats. The level of threats is shown in Fig. 7.

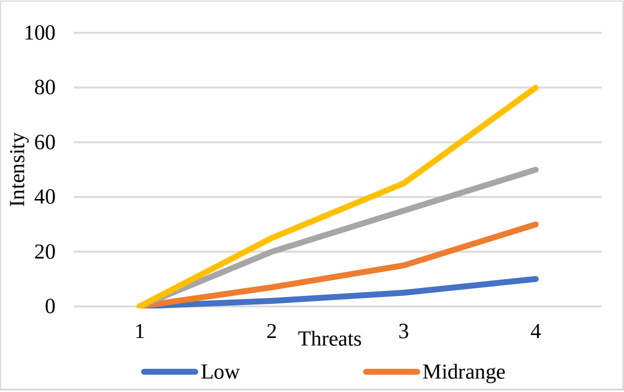
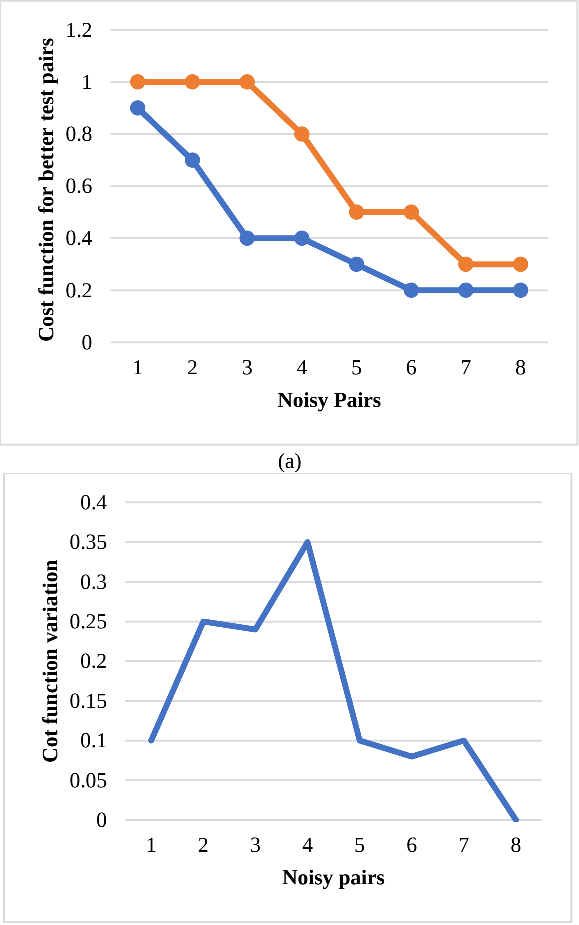


Fig. 7. Threat Ranges.

To produce an N better training pairs as $(|\emptyset_a^{in}\rangle, T|\emptyset_a^{in}\rangle)$, then n destroy by changing them with the training noisy pairs. At each period the changed subgroup is selected arbitrarily. Then, the cost function is measured for all the better test pairs. By select the example, Res-HQCNN [4, $\bar{5}$, 4], with η = 1/1.8 and ε = 0.1 is represented on Fig. 8.



(a)

(b)

Fig. 8. Noisy Training Data Behavior.

In Fig. 8(a), the orange line is the Res-HQCNN results and the blue line are the results, respectively. The Fig. 8(b), plots the cost function variations between the orange lines and blue lines. The x-axis in Fig. 8 demonstrates the number of better training pairs is changed by noisy pairs. Assume a small number of training rounds and pairs, for example, 50 training rounds and 30 training pairs. The cost value for both [4, $\bar{5}$, 4] and [4,5,4] decrease as the amount of noisy pairs raises and the cost value variation is always positive. This demonstrates the [4, $\bar{5}$, 4] superiority for noisy training data with the minimum training rounds and the minimum training pairs. Next, assume the amount of training rounds and pairs are increases as training rounds are 200 and training pairs in 100. Res-HQCNN and QNNs both offer robust toughness to the noisy quantum data when the number of noisy pairings is modest, such as less than 35. The cost values for the orange and blue lines begin to decrease at the same time as the number of noisy pair increases.

When the number of noisy pairings hits 60, the cost variation increases, reaching a maximum when the number of noisy pairs reaches 70. This contains three unstable points (55, -0.0115), (90, -0.0161) and (100, -0.0012) then the variation is negative. There are 21 orange line pairs and blue lines. For every period, the better training data $(|\emptyset_a^{in}\rangle, T|\emptyset_a^{in}\rangle)$, and $(|\emptyset_a^{in}\rangle, |\emptyset_a^{out}\rangle)$ as noisy training data are produced randomly. The $(|\emptyset_a^{in}\rangle\; and\; |\emptyset_a^{out}\rangle)$ elements are casually chosen out a normal spreading before regularization. The training data randomness produces some uneven ideas, it is shown on comparable results. Then, the Res-HQCNN [4, $\bar{5}$, 4] it shows better robustness to the noisy data than [4, 5, 4] QNNs.

To detect the deeper network as [4,$\bar{5}$,$\bar{6}$, 4] to noisy data is shown in Fig. 9. When the amount of training pairs and rounds are minimum like training rounds as 150 and training pairs as 30 on the figure. To notice a sign of improvement from the figure. The cost function variances are always positive. With an increase in the number of noisy pairs, the variation reduces. There is an amount of training rounds and pairs are large, like training rounds as 600 and training pairs 100 in Fig. 10. It's great to have all cost function variances are always positive and there are no unstable points. The greatest variance value is greater than 0.35, but the one in the figure is less than 0.12. This noisy data is deeper as [4,$\bar{5}$,$\bar{6}$, 4] it shows great improvement than [4,$\bar{5}$, 4]. It going via the studies for Res-HQCNN with or without noise and found that it outperforms QNNs in terms of cost function performance. Although this does not exhibit an outcome for Res-HQCNN with the four or more hidden layers, believe that due to the mechanism of its training method, deeper Res-HQCNN would increase cost function performance more.

Fig. 10 and Table I depicts the final results of the presented parameters in the evaluation. To find the encryption size, disentangling size, memory, and execution time as a function of record size. The result shows that as the archive size grows, so does the encryption and unscrambling size. As a result, the execution time grows as well. The result shows that as the archive size grows, so does the encryption and unscrambling size. As a result, the execution time grows as well. In any event, the given paradigm, which differs from various methodologies, secures IoT data in a high-level manner.

Fig. 11 depicts the throughput rate as a function of database size. For every information base size, the QH work provides an ideal level of safety. In QH work, the level throughput is normally excessive, averaging 90%.

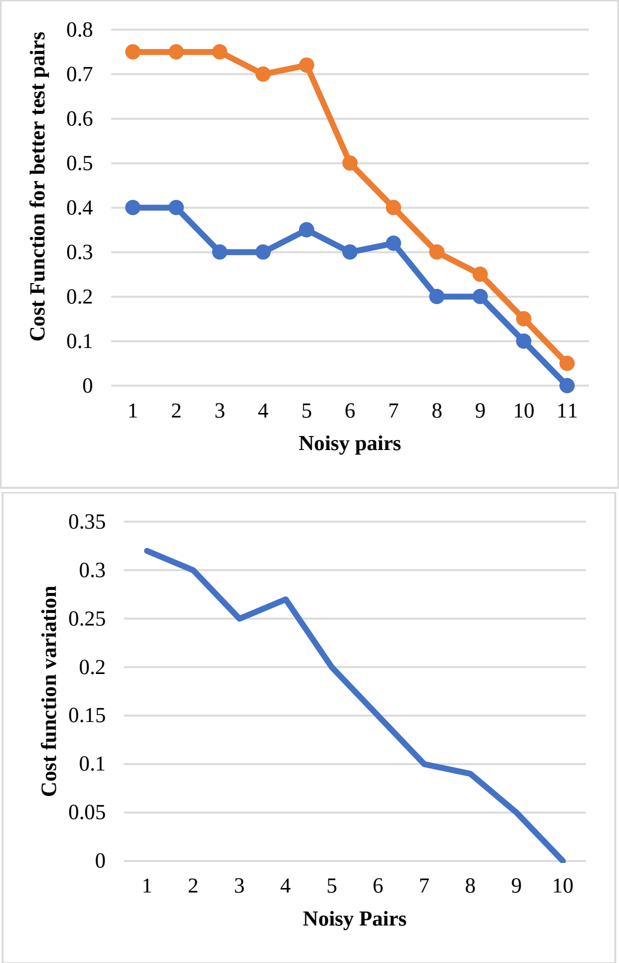


Fig. 9. Deeper Network Detection to Noisy Data.

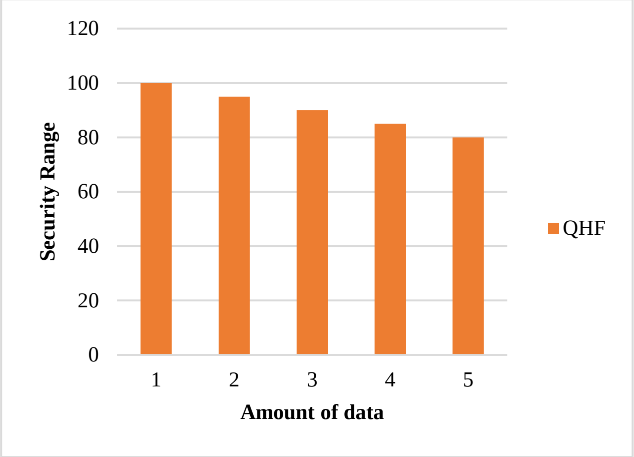


Fig. 10. Amount of Data with Values of Quantum Hash.

TABLE I. QUANTUM HASH FUNCTION

Size of files	Encrypted	Decrypted	Memory	Processing time (ms)
20	27	20	2156432	87231
40	36	40	468769	9423758
60	45	60	476545	10978
80	50	80	576653	113547
100	58	100	563523	115764

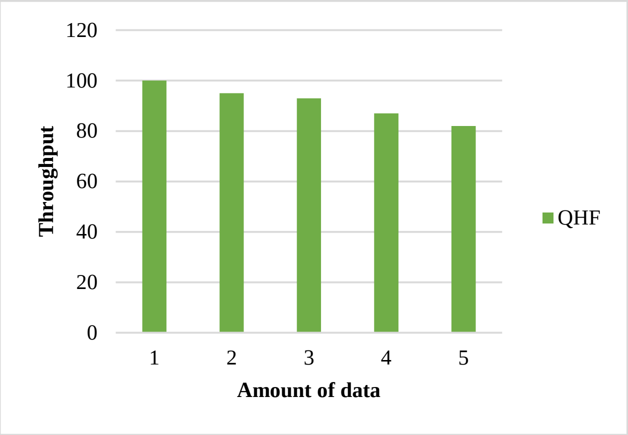


Fig. 11. Amount of Data with the Value of Throughput.

V. CONCLUSION

In this research, to enhance the routine of the cost function for the deeper networks, a quantum-conventional hybrid neural network with deep residual learning was used. Based on the QNNs, a new structure of residual blocks in the quantum concept was developed. Then, the Res-HQCNN training algorithm was also made for different cases. The residual block structure, from the standpoint of information propagation, is similar to the ANN mechanism with deep residual learning in that it permits information to travel from the input layer to any deeper layer. The replications are demonstrated by Res-HQCNN's although it can only work on a regular computer. Due to its non-linear disordered dynamic execution and large key space, quantum hashing work has been proposed as a phenomenal tool for secure IoT communication. The benefits of quantum hashing work have been presented in this research effort as the latest breakthroughs in achieving secure data distribution and information assurance based on Q advancements. For the IDS system, a solution result was modelized using RNN and CNN. It consists of all learning models requested by various network providers. The provided approaches are characterized in terms of increased precision, safety, throughput, and toughness over a few well-known assaults, making them suitable for use in a variety of IoT and cloud applications. In the future, use simulation to investigate the QCNN model, which is more cost effective and has best performance. It is necessary to develop an effective data encoding principle for quantum systems and real information. Finding a way to evaluate threats authorized by offline cyber-attacks is a future suggestion. This study was limited to just online attacks that would be heavily discussed.

ACKNOWLEDGMENT

The authors would like to thank the Deanship of Scientific Research at Jouf University for supporting this work by Grant Code: (DSR-2021-02-0378).

Funding Statement: This work was funded by the Deanship of Scientific Research at Jouf University under grant No (DSR-2021-02-0378).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

REFERENCES

[1] Y. Liang, W. Peng, Z.-J. Zheng, O. Silvén, and G. Zhao, “A hybrid quantum–classical neural network with deep residual learning,” Neural Networks, vol. 143, pp. 133–147, Nov. 2021, doi: 10.1016/j.neunet.2021.05.028.

[2] D. Mu, Z. Guan, and H. Zhang, “Learning algorithm and application of quantum neural networks with quantum weights,” International Journal of Computer Theory and Engineering, vol. 5, no. 5, p. 788, 2013.

[3] S. A. Stein et al., “QuClassi: A Hybrid Deep Neural Network Architecture based on Quantum State Fidelity,” arXiv preprint arXiv:2103.11307, 2021.

[4] T. Michael, “CNN Intrusion Detection for Threat Analysis of a Network,” TURCOMAT, vol. 12, no. 3, pp. 3945–3949, Apr. 2021, doi: 10.17762/turcomat.v12i3.1683.

[5] R. Majumder et al., “Hybrid Classical-Quantum Deep Learning Models for Autonomous Vehicle Traffic Image Classification Under Adversarial Attack,” arXiv preprint arXiv:2108.01125, 2021.

[6] K. Shankar, “Improving the Security and Authentication of the Cloud with IoT using Hybrid Optimization Based Quantum Hash Function,” Feb. 2020, doi: 10.5281/ZENODO.3689761.

[7] Y. Yang, Y. Zhang, G. Xu, X. Chen, Y.-H. Zhou, and W. Shi, “Improving the efficiency of quantum hash function by dense coding of coin operators in discrete-time quantum walk,” SCIENCE CHINA Physics, Mechanics & Astronomy, vol. 61, no. 3, pp. 1–8, 2018.

[8] A. Ajagekar, T. Humble, and F. You, “Quantum computing based hybrid solution strategies for large-scale discrete-continuous optimization problems,” Computers & Chemical Engineering, vol. 132, p. 106630, Jan. 2020, doi: 10.1016/j.compchemeng.2019.106630.

[9] M. Marchi, B. Gharesifard, and P. Tabuada, “Training deep residual networks for uniform approximation guarantees,” p. 12, 2021.

[10] L. Wen, K. Zhou, J. Li, and S. Wang, “Modified deep learning and reinforcement learning for an incentive-based demand response model,” Energy, vol. 205, p. 118019, Aug. 2020, doi: 10.1016/j.energy.2020.118019.

[11] J. Liu, K. H. Lim, K. L. Wood, W. Huang, C. Guo, and H.-L. Huang, “Hybrid Quantum-Classical Convolutional Neural Networks,” arXiv:1911.02998 [quant-ph], Aug. 2021, doi: 10.1007/s11433-021-1734-3.

[12] P. Palittapongarnpim, P. Wittek, E. Zahedinejad, S. Vedaie, and B. C. Sanders, “Learning in Quantum Control: High-Dimensional Global Optimization for Noisy Quantum Dynamics,” Neurocomputing, vol. 268, pp. 116–126, Dec. 2017, doi: 10.1016/j.neucom.2016.12.087.

[13] J. Shi et al., “An approach to cryptography based on continuous-variable quantum neural network,” Scientific reports, vol. 10, no. 1, pp. 1–13, 2020.

[14] S. Hou, G. Yang, and H. Xie, “Optimized initial weight in quantum-inspired neural network for compressing computer-generated holograms,” Optical Engineering, vol. 58, no. 5, p. 053105, 2019.

[15] M. P. Heinrich, M. Stille, and T. M. Buzug, “Residual U-net convolutional neural network architecture for low-dose CT denoising,” Current Directions in Biomedical Engineering, vol. 4, no. 1, pp. 297–300, 2018.

[16] W. Jia, Y. Tian, R. Luo, Z. Zhang, J. Lian, and Y. Zheng, “Detection and segmentation of overlapped fruits based on optimized mask R-CNN application in apple harvesting robot,” Computers and Electronics in Agriculture, vol. 172, p. 105380, 2020.

[17] Y. H. Hwang, “Iot security & privacy: threats and challenges,” in Proceedings of the 1st ACM workshop on IoT privacy, trust, and security, 2015, pp. 1–1.

[18] M. Abomhara and G. M. Køien, “Cyber security and the internet of things: vulnerabilities, threats, intruders and attacks,” Journal of Cyber Security and Mobility, pp. 65–88, 2015.

[19] Y. Hu and X. Lu, “Learning spatial-temporal features for video copy detection by the combination of CNN and RNN,” Journal of Visual Communication and Image Representation, vol. 55, pp. 21–29, 2018.

[20] H. Yu and B. M. Wilamowski, “Levenberg–marquardt training,” in Intelligent systems, CRC Press, 2018, pp. 12–1.

[21] H. F. Lui and W. R. Wolf, “Construction of reduced-order models for fluid flows using deep feedforward neural networks,” Journal of Fluid Mechanics, vol. 872, pp. 963–994, 2019.

[22] D. A. Lorenz and T. Pock, “An inertial forward-backward algorithm for monotone inclusions,” Journal of Mathematical Imaging and Vision, vol. 51, no. 2, pp. 311–325, 2015.

[23] D. Li, Y.-G. Yang, J.-L. Bi, J.-B. Yuan, and J. Xu, “Controlled alternate quantum walks based quantum hash function,” Scientific reports, vol. 8, no. 1, pp. 1–7, 2018.

[24] R. Bernardo-Gavito et al., “Extracting random numbers from quantum tunnelling through a single diode,” Scientific reports, vol. 7, no. 1, pp. 1–6, 2017.

[25] Y.-G. Yang, P. Xu, R. Yang, Y.-H. Zhou, and W.-M. Shi, “Quantum Hash function and its application to privacy amplification in quantum key distribution, pseudo-random number generation and image encryption,” Scientific reports, vol. 6, no. 1, pp. 1–14, 2016.

[26] B. Abd-El-Atty, A. A. Abd El-Latif, and S. E. Venegas-Andraca, “An encryption protocol for NEQR images based on one-particle quantum walks on a circle,” Quantum Information Processing, vol. 18, no. 9, pp. 1–26, 2019.