IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 25 October 2023, accepted 16 November 2023, date of publication 21 November 2023, date of current version 29 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3335625

RESEARCH ARTICLE

Variational Methods in Optical Quantum Machine Learning

MARCO SIMONETTI[1,2], DAMIANO PERRI[2], AND OSVALDO GERVASI[2], (Senior Member, IEEE)
[1]Department of Mathematics and Computer Science, University of Florence, 50134 Florence, Italy
[2]Department of Mathematics and Computer Science, University of Perugia, 06123 Perugia, Italy

Corresponding authors: Marco Simonetti (m.simonetti@unifi.it) and Osvaldo Gervasi (osvaldo.gervasi@unipg.it)

ABSTRACT The computing world is rapidly evolving and advancing, with new ground-breaking technologies emerging. Quantum Computing and Quantum Machine Learning have opened up new possibilities, providing unprecedented computational power and problem-solving capabilities while offering a deeper understanding of complex systems. Our research proposes new variational methods based on a deep learning system based on an optical quantum neural network applied to Machine Learning models for point classification. As a case study, we considered the binary classification of points belonging to a certain geometric pattern (the Two-Moons Classification problem) on a plane. We think it is reasonable to expect benefits from using hybrid deep learning systems (classical + quantum), not just in terms of accelerating computation but also in understanding the underlying phenomena and mechanisms. This will result in the development of new machine-learning paradigms and a significant advancement in the field of quantum computation. The selected dataset is a set of 2D points creating two interleaved semicircles and is based on a 2D binary classification generator, which aids in evaluating the performance of particular methods. The two coordinates of each unique point, x_1 and x_2, serve as the features since they present two disparate data sets in a two-dimensional representation space. The goal was to create a quantum deep neural network that could recognise and categorise points accurately with the fewest trainable parameters possible.

INDEX TERMS Quantum computing, variational methods, deep learning, quantum feed-forward neural networks, optical quantum computing.

I. INTRODUCTION

The phases of human understanding have always alternated between awe at the vastness of the phenomena in front of us and joyous mastery of the goals established. Galileo's insight that mathematics is the simple language that allows us to communicate with nature has been confirmed by the gradual development and introduction of a formidable mathematical arsenal into science. Our faith in positivist determinism at the turn of the 20th century was severely undermined by emerging phenomena that eventually gave rise to quantum mechanics. Machine learning techniques are used to calculate enormous amounts of data [1], [2]. The study of theoretical and practical parallels between particular physical processes and learning systems, such as neural networks, is known as quantum machine learning [3], [4]. Though basic molecular structures can be described by simulating them with automatic calculation tools that require ever-increasing computational power, the observation that deterministic chaos can be generated from models with a seemingly simple apparatus of differential equations and the apparent need suggest that the time has come for a profound reflection on our methods of scientific inquiry.

Feynman claimed that the key to exponentially reducing the computational complexity of the system in issue and adequately regulating the predictive power for the model would be to describe the world around us, which is fundamentally quantum, through a sort of computation based on quantum mechanics. We could also state further that the advent of quantum computing devices would also resolve the issues

The associate editor coordinating the review of this manuscript and approving it for publication was Tao Huang.

associated with the impending passing of the construction limit of present-day computers (Amdahl's law), the potential for a significant reduction in the energy required by current computing devices due to computational reversibility, and the advancement of nanotechnology. Due to such possibilities' potential, there is increasing global interest in such devices and significant investment in the necessary research. As a result, quantum computing devices may answer inquiries that would be impossible for genuine old-style equipment to be quickly processed since they would require the whole time of the universe to do so. That is made possible by various quirks of quantum mechanics that appear at minor scales, such as superposition, entanglement, and interference [5], [6]. Though theoretical research on quantum algorithms points to the prospect of overcoming computer issues that are now insurmountable [7], the technology is still in its infancy and does not yet provide meaningful advantages. The age of Noisy Intermediate-Scale Quantum (NISQ) [8] is still in effect. Implementing a gate-based quantum algorithm is complicated by NISQ-devices' aforementioned limitations on quantum resources and noise. Numerous characteristics of circuit design, including depth, width, and noise, should be taken into account to ascertain if a gate-based algorithm's implementation will function properly on a specific NISQ device [9], [10].

In physics, variational methods are often employed [11], [12], most notably in quantum mechanics [13]. Variational Quantum Algorithms (VQAs), their direct descendants, have emerged as the most powerful method for obtaining a quantum advantage on NISQ devices. Without a doubt, VQAs are the quantum counterpart of powerful machine learning methods like neural networks. Furthermore, as VQAs use parametrised quantum circuits to operate on the quantum computer and subsequently contract out parameter optimisation to a classical optimiser, they make use of the classical optimisation toolkit. Contrary to quantum algorithms created for the fault-tolerant period, this approach has the added advantages of minimising noise and maintaining a light quantum circuit depth [14]. Finding the precise or reasonably close collection of parameter values that minimises a particular cost (loss) function is the goal of this kind of approach. That depends on the parameters themselves as well as, obviously, on the input values, which are the non-trainable portion of the schema. Regulators comprise an output measuring device and a parameter modification circuit. We may argue that the system mimics a traditional physical feedback system (Figure 1), which attempts to minimise a well-defined starting loss function while optimising the set of its parameters.

A more detailed diagram than Figure 1 can be found in Figure 2, where the individual components that make the mentioned transformations possible are highlighted; conceptually, it is possible to divide the system into two parts: a classical information part and a quantum processing part. The latter aims to speed up the calculation and subsequent determination of the final result.

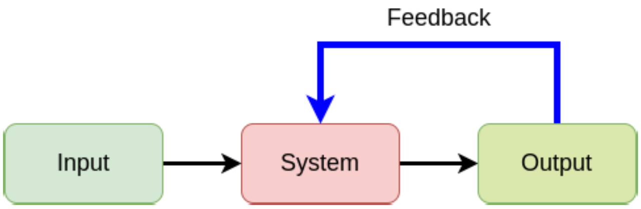


FIGURE 1. Classical feedback system's scheme.

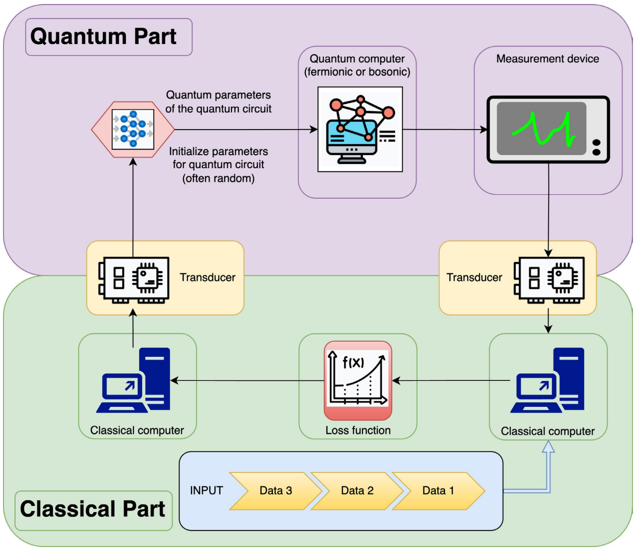


FIGURE 2. General scheme for quantum variational method circuit.

In the classical processing part, a conventional calculator has the purpose of transforming the input data, adapting them to the subsequent quantum processing, and verifying through a function whether the configuration resulting from the quantum calculation is a result of minimum cost concerning a particular function, chosen in the initial phase of analysing the problem.

In the quantum processing part, the appropriately adapted input data modulate the physical parameters of the quantum network; subsequently, the measuring devices detect the eigenstates of the collapsed quantum system, on which the expected value is calculated. This value is returned to the classical part for the processing described above.

Our approach is based on using a *feedforward neural network* (FFNN), usually referred to as a *multilayer perceptron*, whose layers have been built utilising quantum photonic circuits in this particular instance. Eq.1 is the mathematical function that accurately depicts the transformation provided by an FFNN. The fundamental structure, however, is essentially the same: a horizontal stacked multi-layer arrangement with each layer being composed of an initial linear transformation (an *affine transformation*) and a nonlinear function called "activation" (Eq.2)

$$\bar{\mathbf{y}} = f\ (\bar{\mathbf{x}},\ \theta_W,\ \theta_b\ ) = \mathscr{L}_k \circ \mathscr{L}_{k-1} \circ\ \dots\ \circ \mathscr{L}_2 \circ \mathscr{L}_1(\bar{\mathbf{x}}) \quad (1)$$

with $\overline{\mathbf{x}} \in \mathbb{R}^n$ is the input vector, $\overline{\theta_{\mathbf{W}}} \in \mathbb{R}^{p_1}$, $\overline{\theta_{\mathbf{b}}} \in \mathbb{R}^{p_2}$ are the parameter values' vectors and $(k, n, p_1, p_2) \in \mathbb{N}^4$.

$$\begin{aligned} &\mathscr{L}_i : \mathbb{R}^{n_i} \to \mathbb{R}^{n_{i+1}} \\ &\mathscr{L}_i(\overline{\mathbf{x_i}}) = \varphi_i\left(W_i\overline{\mathbf{x_i}} + \overline{\mathbf{b_{i+1}}}\right) \end{aligned} \tag{2}$$

with $i = 1..k$ and $(n_1, n_2, \ldots, n_{k-1}, n_k, n_{k+1}) \in \mathbb{N}^{k+1}$. Moreover

- $W_i \in \mathbb{R}^{n_{i+1} \times n_i}$ is the i-th matrix for the i-th layer, called the *weight matrix* $\left(\text{containing } \overline{\theta_W}\right)$,
- $\overline{\mathbf{x_i}}$ is the i-th input vector,
- $\overline{\mathbf{b_i}} \in \mathbb{R}^{n_{i+1}}$ is the i-th vector, called the *bias vector* $\left(\text{containing } \overline{\theta_b}\right)$,
- φ_i is the non-linear activation function for the *i-th layer*

Usually, we are dealing with a *n-qubits* quantum ansatz, whose transformation is represented by the $n \times n$ unitary matrix U that depends on the vector of parameters θ, with dimensions $m \in \mathbb{N}$ (Eq.1); the loss function we refer to is the *expectation value* measured on every single qubit channel (Eq.3); the observable is the operator O_i, where Z_i is the Z-Pauli Operator, acting on i-th qubit (Eq.4); the quantum state on which to operate the measurements will be the state resulting from the ansatz (ψ in Eq.3).

$$\left|\psi\left(\theta\right)\right\rangle = U\left(\theta\right) \cdot |0\rangle^{\otimes n} \tag{3}$$

$$O_i = Z_i\,, \quad i \in \{1..n\} \subset \mathbb{N} \tag{4}$$

$$\left\langle E_i\left(\theta\right)\right\rangle = \left\langle \psi^\dagger\left(\theta\right) \middle| O_i \middle| \psi\left(\theta\right)\right\rangle \tag{5}$$

The ability of quantum computers to resolve particular problems more quickly than classical ones is widely recognised. Nevertheless, packing data into a quantum computer is not so simple because the information should be encoded as quantum bits. They may interact with the data in various ways, making a wide range of information encodings possible [15]. There are several ways to incorporate data; the most popular ones are Basis Encoding [16], [17], Amplitude Encoding [18], [19], [20], and Angle Encoding [21], [22].

However, an alternative approach may be used to achieve the same results: an optical-quantum layer-based FFNN [23]. Thus, the following transformation (Eq.6) can be ensured by using the Singular Value Decomposition (SVD - Eq.7):

$$f : \left|\,\overline{\mathbf{x}}\,\right\rangle \to \left|\,\varphi\left(W\,\overline{\mathbf{x}} + \overline{\mathbf{b}}\right)\right\rangle \tag{6}$$

with $\left|\,\overline{\mathbf{x}}\,\right\rangle = \bigotimes_{i=1}^{n} |x_i\rangle$, n is the dimension of the feature space and $\overline{\mathbf{x}}$ the generic feature entering the system. Data can be encoded in position eigenstates.

The Singular Value Decomposition theorem, which guarantees the factorisation of W into three matrices, two orthogonal and one as a positive diagonal matrix, may be used to decompose the matrix itself. That ensures we can utilise specific quantum gates that can mimic the behaviour determined by the corresponding matrices.

$$W = U \cdot \Sigma \cdot V^T \tag{7}$$

with

- $U^{n\times n}(\mathbb{R}) \mid \; U^{-1} = U^T$
- $\Sigma^{n\times m}(\mathbb{R}) \mid \; \Sigma = diag(\sigma_1, \sigma_2, \ldots, \sigma_p)$
- $V^{m\times m}(\mathbb{R}) \mid \; V^{-1} = V^T$
- $(m, n) \in \mathbb{N}^2$

Regarding the physical implementation, squeezer gates may be utilised to obtain the diagonal transformation, while interferometers (a combination of beamsplitters and rotation gates) can be employed to achieve the orthogonal ones. The addition operation using the bias vector, $\overline{\mathbf{b}}$, is then obtained by appending position displacement gates. Hanging a Kerr gate as the final circuital block might be a potential option to produce a non-linear transformation; this is the most common choice. Figure 3 shows the circuit diagram of a single layer relating to the input use of a single q-mode, while Figures 4 and 5 represent the situations with two q-modes and four q-modes, respectively. Our research intends to investigate several facets of quantum computation's applicability to machine learning. Concerning learning a straightforward classification task, we specifically compared the abilities of a fundamental OQ-FFNN (Optical Quantum FFNN) with dense layers with an identical network only made up of photonic quantum components.

The aim of this work is the development of a quantum deep neural network able to recognise and classify points accurately with the fewest trainable parameters possible.

Furthermore, this investigation intends to continue a study theme initiated in one of our previous papers, in which the identical classification issue was solved using a conventional quantum network [24].

II. RELATED WORKS

Since 2010, there has been a growing interest in applying machine learning techniques [25] to quantum computing [26], [27], [28]. Early attempts to efficiently simulate the quantum world were prosperous to the extent that small physical systems [29], [30] (with few particles) were considered; the use of high-performance computers was necessary due to the large dimensions involved [31], [32], [33], [34], [35]. The introduction of virtual and augmented reality to better explain the concepts of quantum mechanics is interesting from a didactic point of view [36]. Quantum machine learning [37], [38], [39] integrates quantum algorithms into machine learning programmes. The phrase is most frequently used to describe machine learning algorithms that analyse classical data and are executed on a quantum computer. Quantum machine learning uses qubits, quantum processes, or specialised quantum systems to speed up computation [40].

A wide range of concepts with varying degrees of similarity to conventional neural networks are included in current QNN proposals [41], [42], [43]. The challenge of integrating the linear and non-linear parts and the unitary framework of quantum mechanics is at the heart of quantum neural network theory. Quantum neural networks are only one form of the most recent type of machine learning models implemented on quantum computers. They broadly

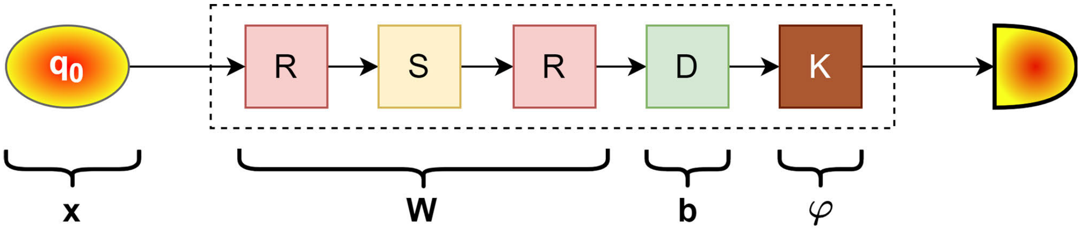


FIGURE 3. Circuital block diagram for a single layer, single q-mode, related to an optical-quantum layer-based FFNN.

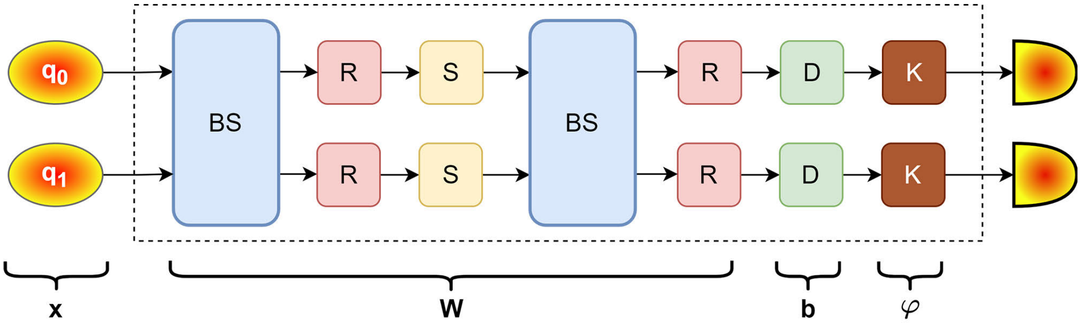


FIGURE 4. Circuital block diagram for a single layer, two q-modes, related to an optical-quantum layer-based FFNN.

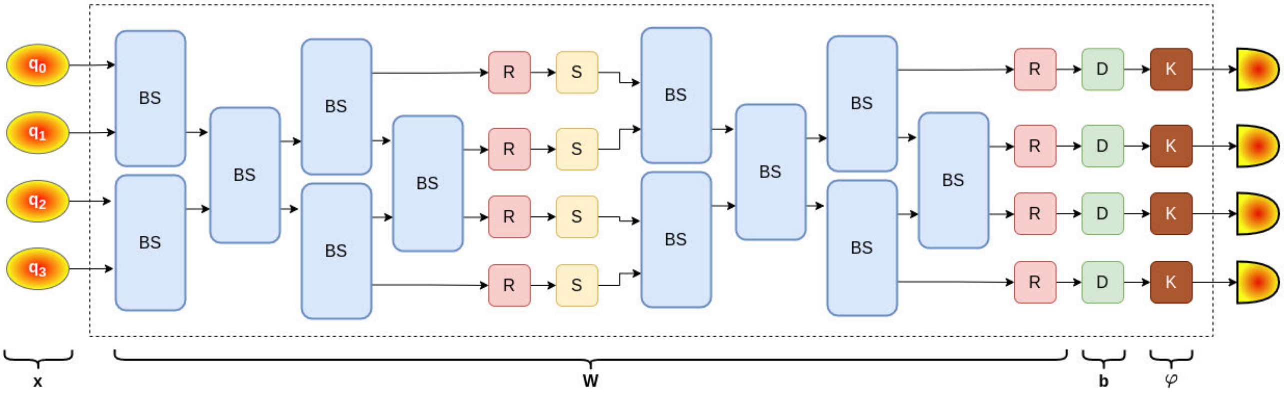


FIGURE 5. Circuital block diagram for a single layer, four q-modes, related to an optical-quantum layer-based FFNN.

use quantum phenomena like superposition, entanglement, and interference to exploit possible benefits such as quicker training and processing [44], [45], [46].

It has recently been proposed that the representation of quantum information need not be binary or discrete. Instead, it is also possible to leverage the innately "continuous" quantum features of matter, which would inevitably result in encoding the information in continuous variables (CV). The position and momentum of a particle are typical examples [47], [48], [49], [50]. It needs a quantum circuit with a universal layer structure so that we can manufacture any CV state with no more than polynomial complexity in order to do arbitrary proper transformations for the learning process by the machine [51], [52]. Therefore, the architecture to be selected must be composed of layers, with parameterised Gaussian and non-Gaussian gates present in each layer [53], [54], [55]. The non-Gaussian gates provide the model with both nonlinearity and universality [56], [57], [58]. The application of photonic quantum machine learning is proving effective in both traditional classification and regression issues, which is undoubtedly a fascinating finding [59], [60], [61], [62]. However, more importantly, it is fast improving our knowledge of quantum processes themselves [63], [64].

III. THE SYSTEM ARCHITECTURE

We tested three different types of OQFFNN, whose general structure is identical, and the differentiation comes from the distinct implementations of the quantum networks. All our models consist of a series of quantum layers ($\mathscr{L}_i$) followed by a measurement apparatus whose outputs go into a classic dense layer that enables the binary categorisation of the input items (Figure 6). The Number Operator ($\hat{n}_i = \hat{a}_i^\dagger \hat{a}_i$, that is the sequential action of the annihilation and creation operators) has been selected as the observable in our experiments, and the measure will be its average value, $\langle \hat{n}_i \rangle$, for each i-th q-mode.

The chosen dataset is a set of 2D points, so it can be represented as a matrix whose shape is (*number_of_features*, 2).

A. IMPLEMENTED QUANTUM NETWORKS

We can now examine the three varieties of quantum layers we implemented.

1) FIRST KIND OF NETWORK

The dataset features are entered into the network as values for the two parameters (ρ, φ) of a classical coherent state $|\alpha\rangle$, with $\alpha \in \mathbb{C} \mid \alpha = \rho \cdot e^{i \cdot \varphi}$. In order to allow a direct correspondence between the coordinates of the point and the geometric meaning of the parameters of the coherent state given in input, the pair of coordinates (x_1, x_2) that identify a point of the plane in an orthogonal Cartesian reference are transformed into polar coordinates (ρ_1, θ_1) before being transferred as input into the network. The quantum layer (Figure 7) is composed of a Rotational Gate (R), a Squeezing Gate (S), another Rotational Gate (R), a Displacement Gate (D) and a Kerr Gate (K).

2) SECOND KIND OF NETWORK

The features of the input dataset are represented in the network as the phases φ of an equal number of coherent states, whose amplitude ρ is arbitrarily set to 1, to improve the numerical simulation's efficiency without straying too far from the typical values of a physical implementation. The quantum layer (Figure 8) is composed of two BeamSplitters (BS), four Rotational Gates (R), two Squeezing Gates (S), two Displacement Gates (D) and two Kerr Gates (K).

3) THIRD KIND OF NETWORK

The dataset features are entered into the network as values for the two parameters (ρ, φ) of a two-mode squeezing gate, whose inputs are Vacuum States. The quantum layer (Figure 9) is composed of four Rotational Gates (R), two single Squeezing Gates (S), two Displacement Gates (D) and a Cross-Kerr Gate (K).

IV. DATA EXTRACTION AND PROCESSING

The *two-moons* database, which creates two interleaving semicircles of 2D points and is typical for the study of clustering and classification techniques, was chosen to solve the classification challenge.

The Python Scikit-Learn library's[1] dedicated function was used to create the dataset. The input domain is $[-2.0, 3.0] \times [-2.0, 2.5] \subset \mathbb{R}^2$. One thousand five hundred samples were randomly generated and uniformly distributed over the two classes.

The number of samples in the dataset was divided in such a way as to ensure the following quotas: 75% for the training set, 15% for the validation set and the remaining 10% for the test set (Figure 10).

Moreover, the features have been given as points of a plane in polar coordinates (Figure 11) to improve the operation of particular gates whose parameters operate on complex values (in $\mathbb{C}$ set). The kind of input in the Cartesian or Polar form shall be defined for each model in section V.

V. DISCUSSION OF RESULTS

All the models to be discussed have been implemented using the known open-source Python library, PennyLane,[2] with StrawberryFields[3] backend for simulation, both by Xanadu [65], [66]. Both can be perfectly interfaced with the most famous Deep Learning frameworks, such as Keras, TensorFlow and PyTorch.

All tests and simulations were performed using the specialised open-source library for Deep Learning, Keras.[4]

Every model has a traditional dense layer as the last layer (Figure 6), which serves as a classifier. Its input is the average number of photons processed by the quantum network, and its output is a float value between 0 and 1. The sigmoid function is utilised as its activation function. All the models were compiled using:

- *optimizer*: SGD (Stochastic Gradient Descent), with a learning rate equal to 0.01
- *loss function*: binary cross-entropy
- *metric*: accuracy

1) MODEL 1

The simulations were conducted initially using a simple structure, consisting of a single quantum layer, for a total of 9 parameters (Figure 12). Indeed, the number of parameters to be trained for the first model is given by 2 (parameters for the final Dense layer) and the product between the number of quantum layers to be used and 7 (the sum of the parameters of the single quantum gates making up the layer - see Appendix A for details).

Figures 13 and 14 show the metrics (LOSS: loss function on training set, ACCURACY: accuracy on training set, VAL_LOSS: loss function on validation set, VAL_ACCURACY: accuracy on validation set) for the Model n.1, trained for 50 epochs. The outcomes are superb. As shown in Figures 13 and 14, it is evident that the search algorithm has found the best solution, enabling the model to correctly classify all of the data, both the samples of the training set and of the validation set, as early as the 46th epoch. Specifically, Figure 14 represents the attempts of the

[1] https://scikit-learn.org/stable/index.html

[2] https://pennylane.ai/

[3] https://strawberryfields.ai/

[4] https://keras.io/

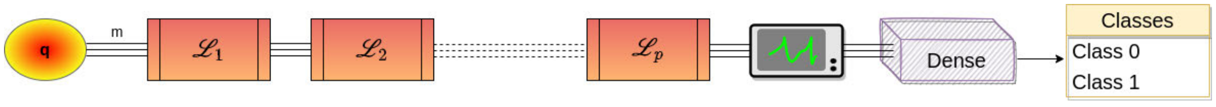


FIGURE 6. Common structure for the proposed OQFFNN (m is the input dimension).

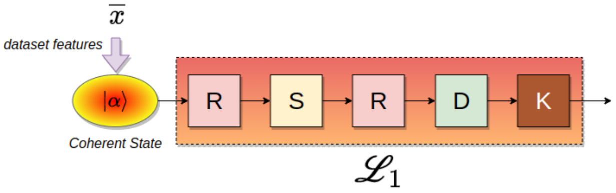


FIGURE 7. First type of layer.

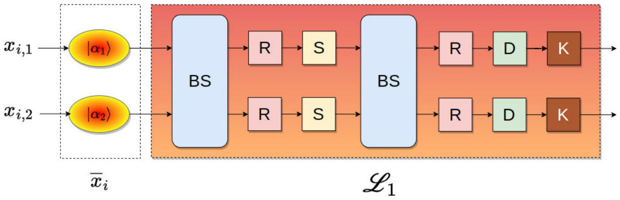


FIGURE 8. Second type of layer.

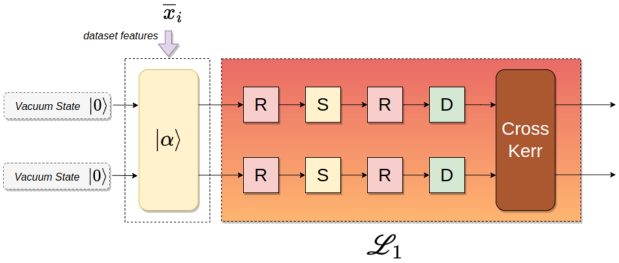


FIGURE 9. Third type of layer.

model to adjust the accuracy to the validation set; the initial natural oscillations gradually dampen after many epochs of training, showing the convergence of the model itself. The optimal parameters' values[5] are displayed in Tab.1, while for the dense layer are $(w, \quad b) = (-11.58, \quad 6.2)$. The results provided in Figure 15 indicate what happens when the model is applied to the test set (150 samples), confirming the model's excellent performance (Test Accuracy: 100%).

Other simulations were run using more quantum layers, but none of them showed a discernible improvement over the single-layer experiment. Specifically, we conducted three simulations using respective 2, 3, and 4 layers.

2) MODEL 2

The number of parameters to be trained for the model n.2 is given by 3 (parameters for the final Dense layer) and the product between the number of quantum layers to be used and 16 (the sum of the parameters of the single quantum gates making up the layer). Following the reasoning shown in section I (Introduction), it was decided to utilise phaseless beamsplitters in this simulation so that each item in its representation matrix might have a real value.

The simulations were conducted initially using a simple structure, consisting of a single quantum layer, for a total of 19 parameters (Figure 16).

Figures 17 and 18 show the metrics (LOSS: loss function on training set, ACCURACY: accuracy on training set, VAL_LOSS: loss function on validation set, VAL_ACCURACY: accuracy on validation set) for the Model n.2, trained for 50 epochs.

The outcomes are satisfactory. The optimal parameters' values[6] are displayed in Tab.2, while for the dense layer, they are $w = (-7.64, \quad 6.90)^T$ and $b = -0.33$. The results provided in Figure 19 indicate what happens when the model is applied to the test set (150 samples), confirming the model's good performance (Test Accuracy: 96.67%).

By starting with this model, it is feasible to increase the degree of categorisation and moderate the oscillations during the solution research phase. The phase angle parameters will be added to the individual BeamSplitters, taking the reflectivity values from real to complex. The number of training parameters for the quantum layer is 18; the final number is 21.

Figures 20 and 21 show the usual metrics for the Model n.2, with phase angles, trained for 50 epochs. The values of the best parameters[7] determined after network training are displayed in Table 3.

We have got the following values for the dense layer's parameters: $w = (-8.04, \quad 7.08)^T$ and $b = -0.18$. The Test Set's scores (Figure 22) are likewise quite good, with an accuracy performance of 97.33%.

It was noticed that the model's performance did not meaningfully improve with the addition of more quantum layers to the network.

3) MODEL 3

A hybrid of the first two, the third kind of layer (Figure 9) contains two q-modes but does not employ beam splitters. The interaction between states is achieved by using a Two-Squeezing Gate, which serves the function of suitably encoding and modulating the input information, while the

[5] Coefficient k is in mV^{-2}.

[6] Coefficient k is in mV^{-2}.

[7] Coefficient k is in mV^{-2}.

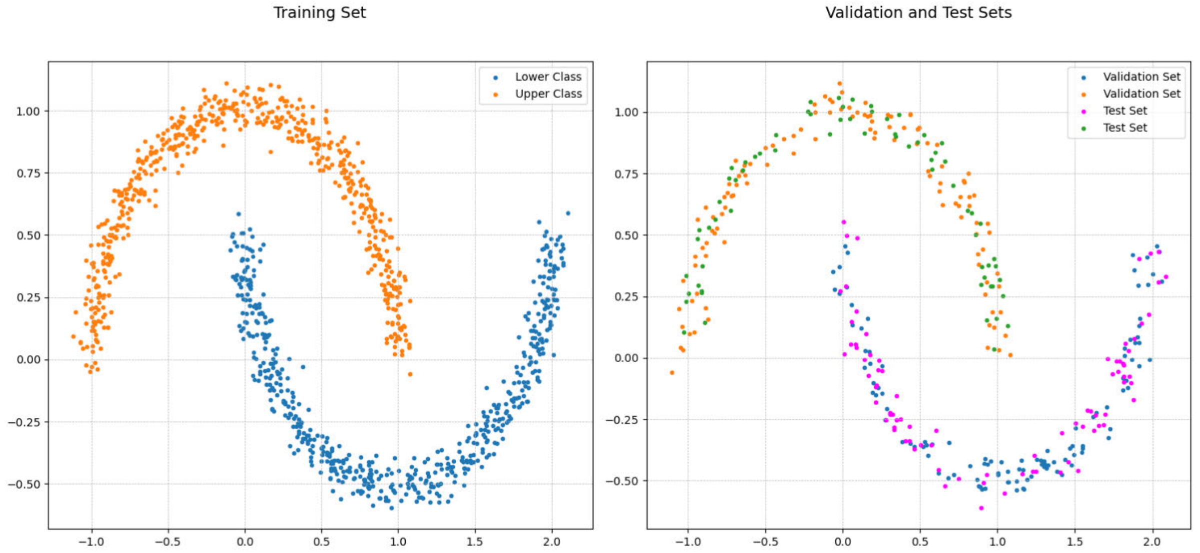


FIGURE 10. **Dataset: Distribution of pattern points on the plane.**

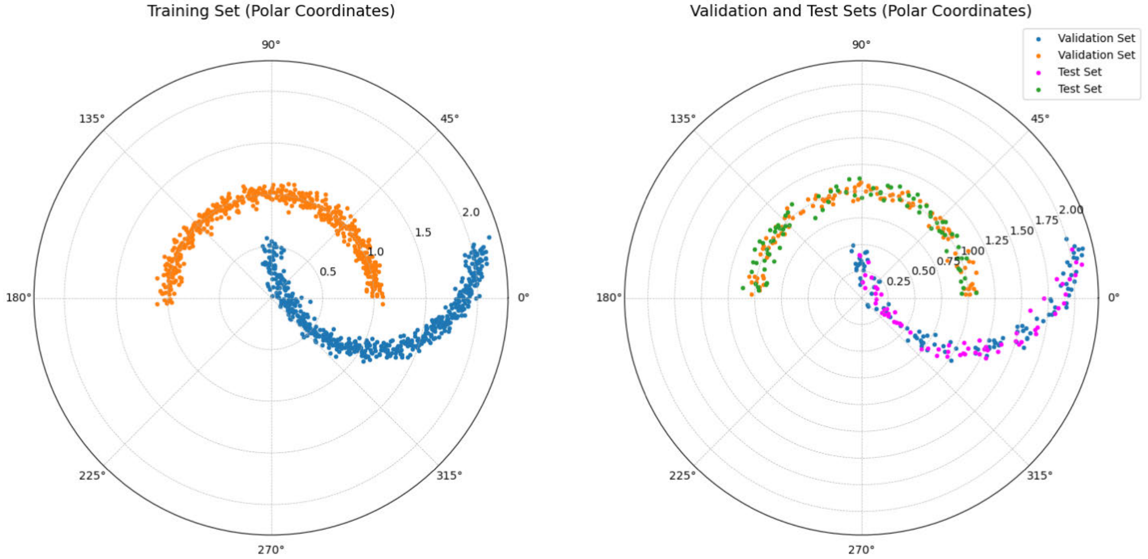


FIGURE 11. **Dataset: Distribution of pattern points on the plane in polar form.**

Cross-Kerr Gate is found in the final step of the Quantum Layer. They both carry out nonlinear transformations commonly utilised to generate entanglement in quantum models with continuous variables.

The simulations were conducted initially using a simple structure, consisting of a single quantum layer, for a total of 16 parameters (Figure 23): 13 belong to quantum layer.

Figures 24 and 25 show the metrics (LOSS: loss function on training set, ACCURACY: accuracy on training set, VAL_LOSS: loss function on validation set, VAL_ACCURACY: accuracy on validation set) for the Model n.3, trained for 50 epochs. The outcomes are satisfactory. The optimal parameters' values[8] are displayed in Tab.4, while for the dense layer, they are $w = (-6.13, -7.14)^T$ and $b = 7.43$. The results provided in Figure 26 indicate what happens when the model is applied to the test set (150 samples), confirming the model's good performance (Test Accuracy: 89.33%).

Further investigations showed that increasing the number of layers leads to a partial but meaningful improvement in performance:

[8] Coefficient k is in mV^{-2}.

TABLE 1. Optimised parameters' set for the quantum network.

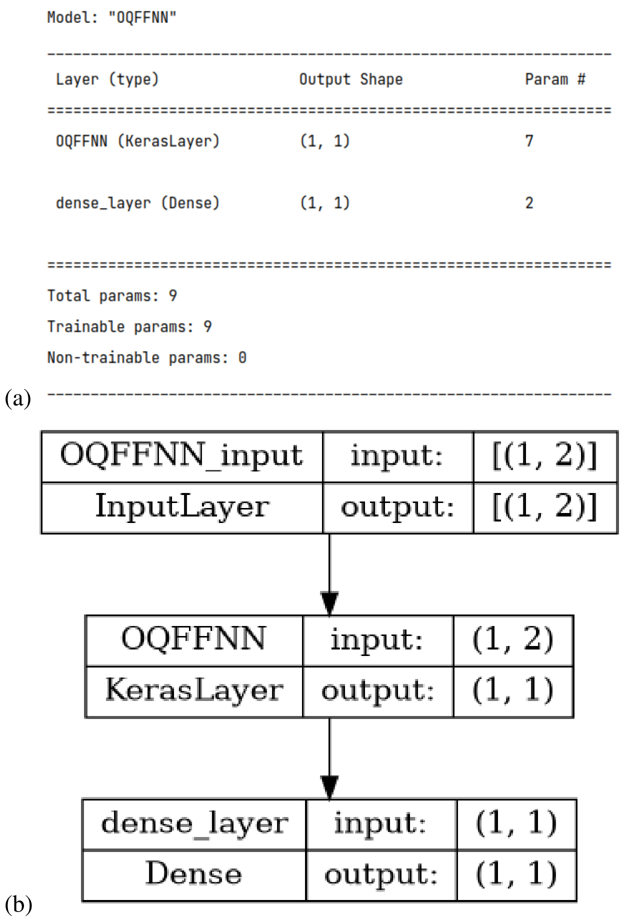
R	S		R	D		K
φ	ϱ	φ	φ	ϱ	φ	**k**
1.28	0.46	0.64	2.56	0.28	0.06	$3.2 \cdot 10^{-7}$

TABLE 2. Optimised parameters' set for the quantum network (Model without phase angles for beamsplitters.)

	BS		R	S		BS		R	D		K
Mode	ϑ	φ	φ	ϱ	φ	ϑ	φ	φ	ϱ	φ	**k**
0	1.59	0.00	2.43	0.07	0.00	0.12	0.00	0.06	0.95	3.85	0.30
1			0.39	0.47	0.40			0.90	0.08	0.02	$5.32 \cdot 10^{-8}$

TABLE 3. Optimised parameters' set for the quantum network (Model with phase angles for beamsplitters.)

	BS		R	S		BS		R	D		K
Mode	ϑ	φ	φ	ϱ	φ	ϑ	φ	φ	ϱ	φ	**k**
0	0.03	1.29	1.76	0.42	0.04	0.14	0.05	0.02	0.03	0.86	$5.45 \cdot 10^{-8}$
1			0.44	0.24	0.02			0.00	0.27	2.23	$6.64 \cdot 10^{-7}$



```
Model: "OQFFNN"

_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 OQFFNN (KerasLayer)         (1, 1)                    7


 dense_layer (Dense)         (1, 1)                    2


=================================================================
Total params: 9
Trainable params: 9
Non-trainable params: 0
_________________________________________________________________
```

FIGURE 12. (a) Number of training parameters; (b) Model 1 - layers' scheme.

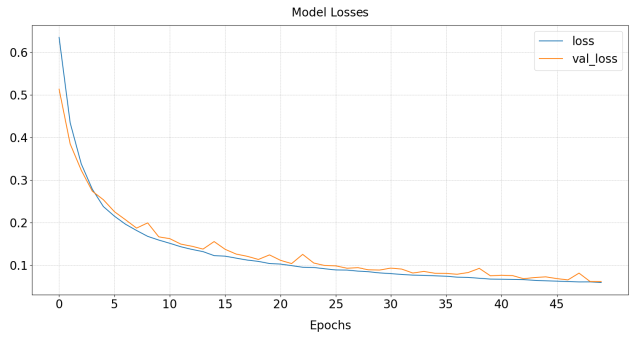


FIGURE 13. Loss and validation loss for model 1.

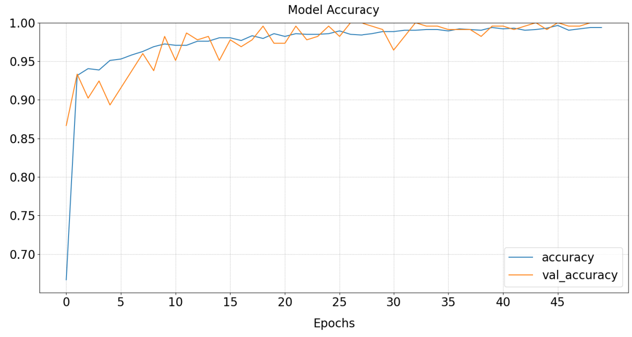


FIGURE 14. Accuracy and validation accuracy for model 1.

- Model 3, with 2 quantum layers
  - Total trainable parameters: 29
  - Test Accuracy: 91.33%
  - ROC Curve Area: 0.90
- Model 3, with 4 quantum layers
  - Total trainable parameters: 55
  - Test Accuracy: 99.33%
  - ROC Curve Area: 0.99
- Model 3, with 8 quantum layers
  - Total trainable parameters: 107
  - Test Accuracy: 100.00%
  - ROC Curve Area: 1.0

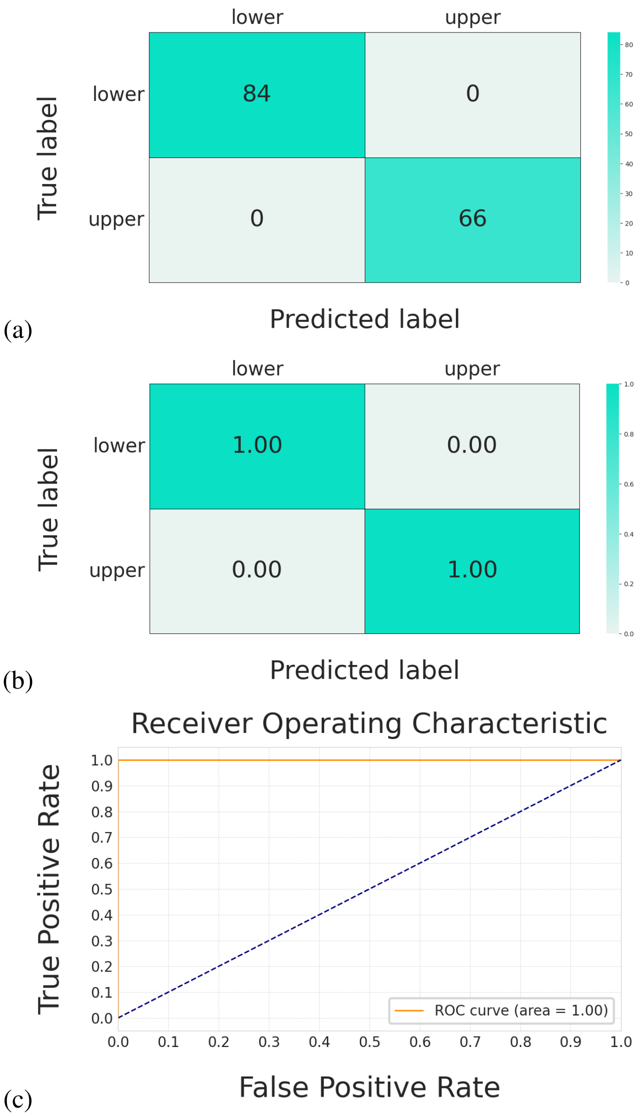
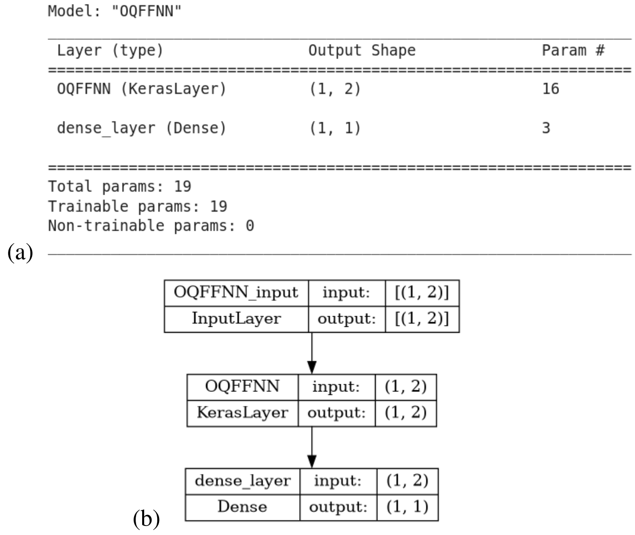


FIGURE 15. (a) Confusion matrix for test set; (b) Normalised confusion matrix; (c) ROC curve.

Furthermore, replacing the Cross-Kerr layer with a Two-Squeezing Gate yields an excellent result after 50 epochs of training (Figure 27), hitting 100% in the Accuracy Test, with only 17 trainable parameters.

So, the research presented in this chapter aimed to delve deeper into the feasibility of using quantum computers for machine learning purposes in the NISQ era by attempting to train a simple network consisting of photonic quantum gates. The study examined various essential aspects of a statistical classifier, such as the convergence of the proposed model, the accuracy of the result concerning a standard metric, the minimisation of the number of trainable parameters (i.e., the number of quantum gates required in the implemented circuit), and the scalability of the solution. The results obtained from this research and other similar recent studies [67] suggest that quantum computation can be used to advantage in the field of Machine Learning.


```
Model: "OQFFNN"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 OQFFNN (KerasLayer)         (1, 2)                    16

 dense_layer (Dense)         (1, 1)                    3

=================================================================
Total params: 19
Trainable params: 19
Non-trainable params: 0
_________________________________________________________________
```

FIGURE 16. (a) Number of training parameters; (b) Model 2 - layers' scheme.

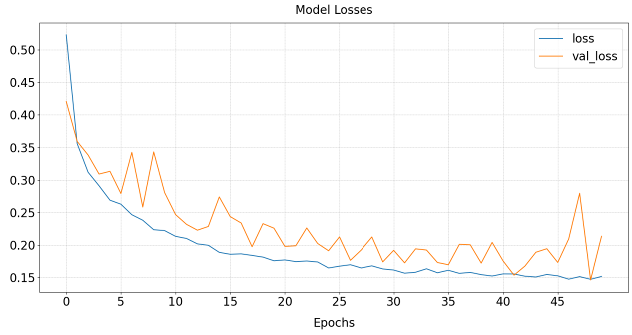


FIGURE 17. Loss and validation loss for model 2.

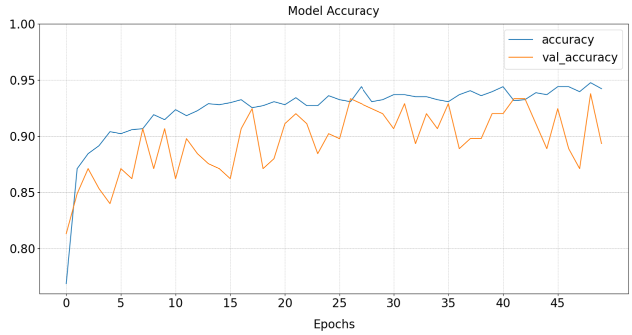


FIGURE 18. Accuracy and validation accuracy for model 2.

In a previous works [24] , the solution was a Fermionic Quantum Machine Learning that leveraged two different libraries to simulate the proposed network, which comprised rotation gates and Cnot. It produced consistent results as the noise level on the dataset changed. Only twenty epochs were needed to effectively train the network, which achieved a remarkable 100% recognition rate for samples belonging to the *Validation Set* when the noise level was low. Moreover,

TABLE 4. Optimised parameters' set for the quantum network.

	R	S		R	D		CK
Mode	φ	ϱ	φ	φ	ϱ	φ	k
0	0.24	0.0	0.05	0.31	0.72	0.01	$1.58 \cdot 10^{-12}$
1	0.24	0.001	0.0	0.19	0.22	0.03	

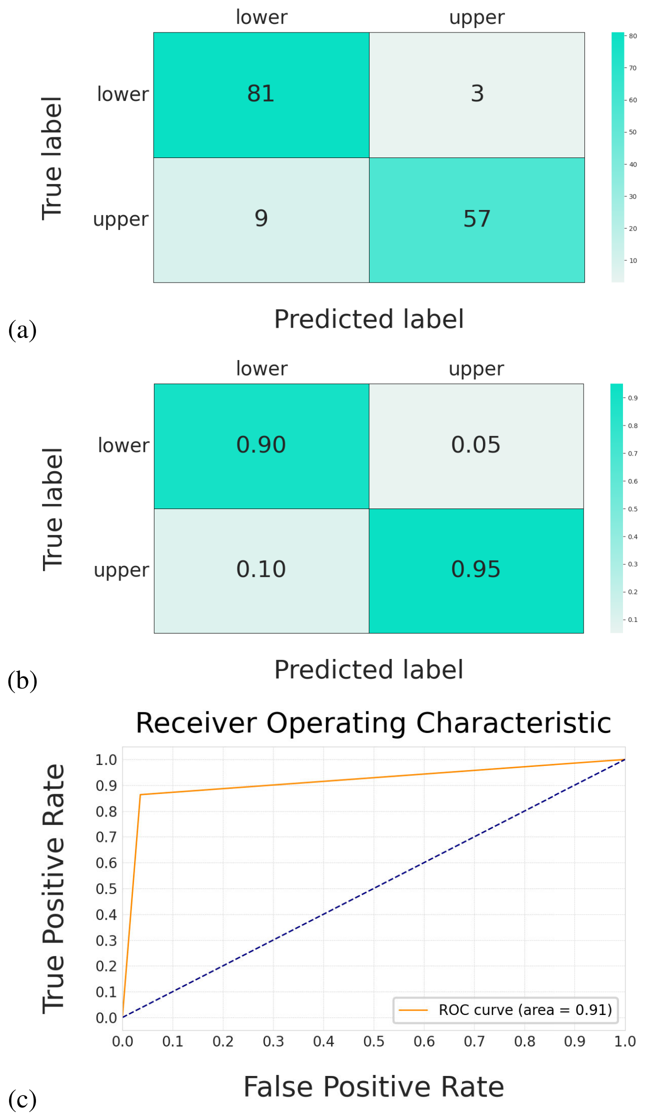


FIGURE 19. (a) Confusion matrix for test set; (b) Normalised confusion matrix; (c) ROC curve.

it achieved an impressive 85% recognition rate for too high noise levels. The progress of the Loss and Accuracy curves of the model suggested that it was robust to noise and immune to the drop-out phenomenon. Surprisingly, a similar network for classical Machine Learning (e.g., a simple FFNN) needed 354 trained weights (although relatively few for networks of this type) versus the 54 of the proposed solution. The degree of scalability was excellent, as the size of the individual dataset did not tie to the network structure.

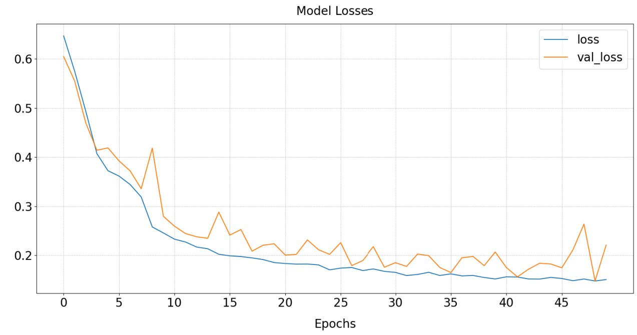


FIGURE 20. Loss and validation loss for model 2 with phases.

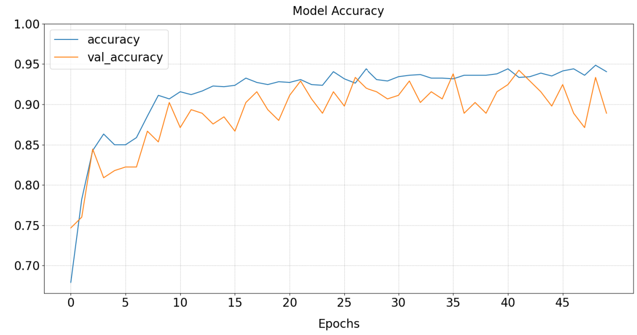


FIGURE 21. Accuracy and validation accuracy for model 2 with phases.

The most notable disadvantage during simulations was the high run-time, but typical of machine learning hybrid solutions (Classic + Quantum). In contrast, in this research, the primary constituents of the quantum circuit are linear and non-linear optical components. The elements of the dataset modulate the incoming photons, transforming them into specific prepared quantum states. Here, the goal, besides the model's convergence and classification ability, was to limit the number of trainable parameters while retaining an accuracy of more than 80%. The three models had surprisingly comparable patterns, confirming their quality and validity and proving their efficacy. This investigation, which employed quantum mechanics, attempted to compare the behaviours emerging from several circuit implementations of the kernel method on a not overly complicated dataset. The results are excellent for all the models investigated, demonstrating the circuit structures' effectiveness. Some

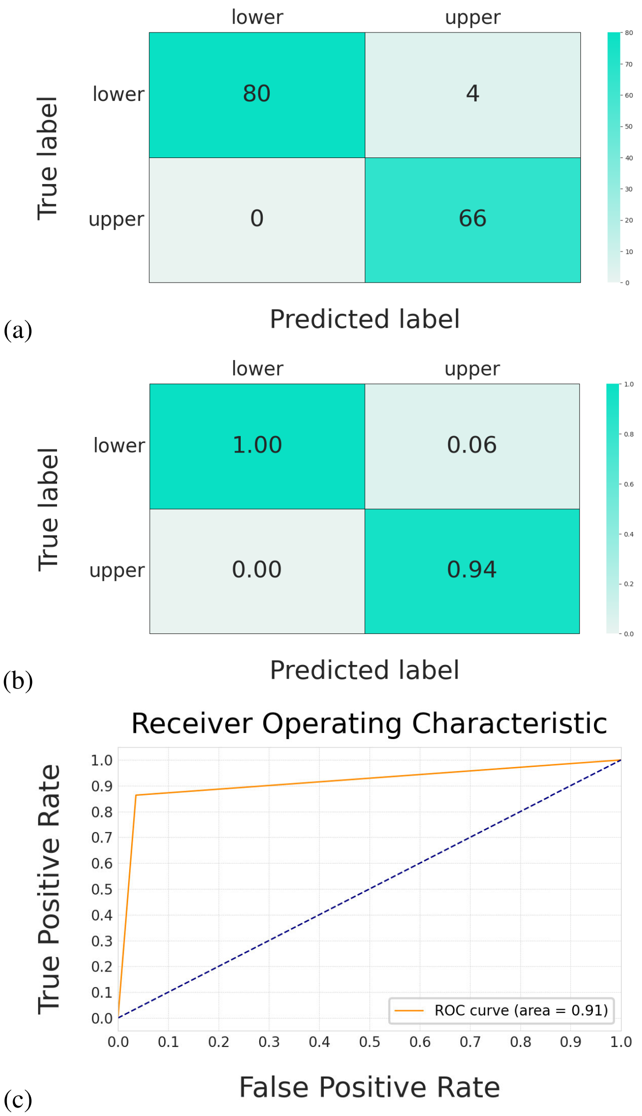
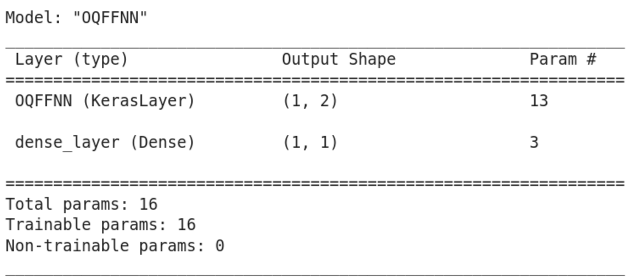


FIGURE 22. **(a) Confusion matrix for test set; (b) Normalised confusion matrix; (c) ROC curve.**



```
Model: "OQFFNN"

 Layer (type)                Output Shape              Param #
=================================================================
 OQFFNN (KerasLayer)         (1, 2)                    13

 dense_layer (Dense)         (1, 1)                    3

=================================================================
Total params: 16
Trainable params: 16
Non-trainable params: 0
```

FIGURE 23. **Number of training parameters.**

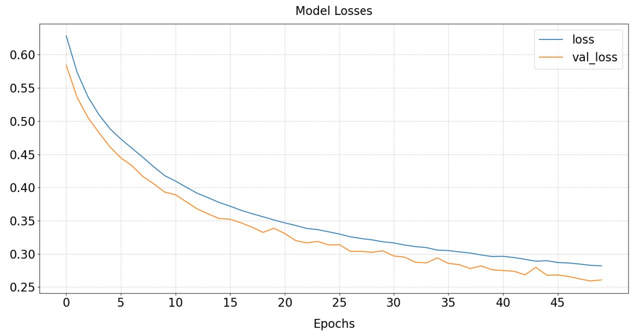


FIGURE 24. **Loss and validation loss for model 3.**

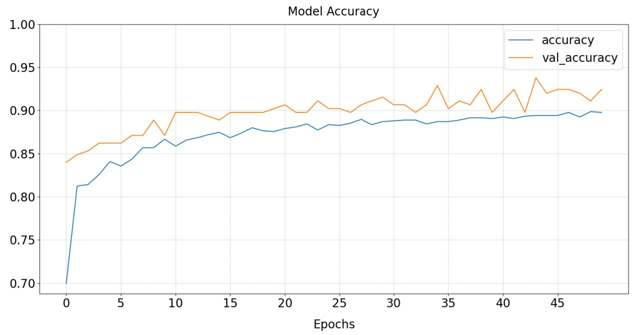


FIGURE 25. **Accuracy and validation accuracy for model 3.**

models have achieved accuracy levels of 100%, while others have never dropped below 90%, exhibiting classification robustness far beyond expectations.

The first model is straightforward: it has seven trained parameters. It receives in input a coherent state, the complex components modulated by the two input values of the data vector. It has an excellent answer: Once trained, it can recognise all new samples proposed with a 100% Test Accuracy.

The second model receives in input two consistent states, the phase of which is modulated by the input data. The confusion matrix and the ROC curve highlight a good model response, with a recognition accuracy of the test data of 91%.

The third model has a very similar structure to the second. Still, the interaction between the two photons in the final stage reduces the performance to 88%, a very high value, considering the low number of trained parameters (equal to 13). An excellent improvement is achieved by replacing the Cross-Kerr stadium with a Two-Squeezing Gate: the results obtained are comparable to those of Model 1.

The limit of the first model is its inability to be scalable if taken in its original version, as its input is a coherent state that depends solely on two parameters. Therefore, datasets with many features cannot be directly encoded in the network. The second and third models are more scalable than the first one. In these models, each input photon contains information about every feature in the dataset. However, minimising the number of gates in the structure is essential to ensure high performance. That is because other stages that stack horizontally can significantly degrade the model's performance.

The research's aim to be done shortly will concern the capacity to categorise increasingly complicated and

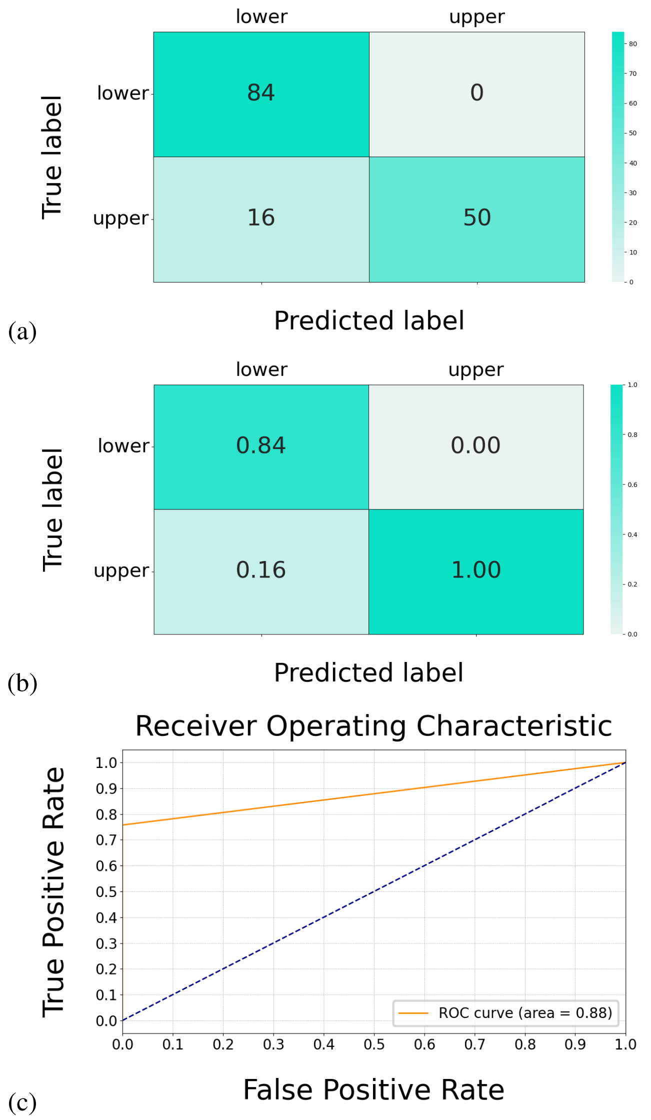


FIGURE 26. **(a) Confusion matrix for test set; (b) Normalised confusion matrix; (c) ROC curve.**

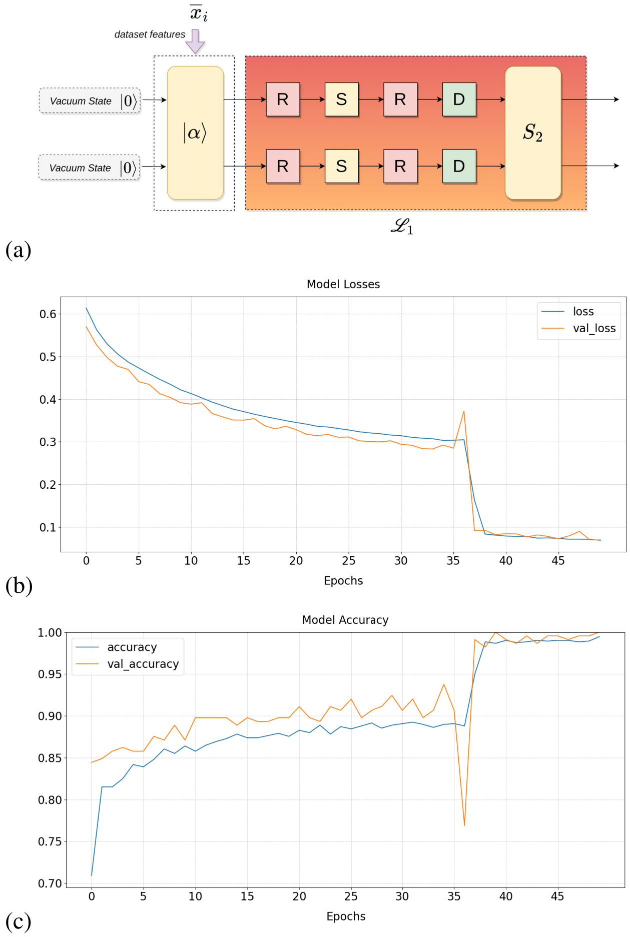


FIGURE 27. **(a) An interesting variant for Model 3; (b) Loss and Validation Loss; (c) Accuracy and validation accuracy.**

structured data sets while leaving a suitably low number of gates needed to accomplish the task.

VI. CONCLUSION

Variational circuits play a crucial role in the field of hybrid algorithms, serving as a link between classical and quantum computing. These circuits leverage a parameterised design that can be adjusted to achieve a specific outcome, like tuning a quantum circuit to simulate particular input-output relationships based on training data.

However, despite the apparent simplicity of this concept, the actual execution of variational circuits is incredibly complex. One of the most significant challenges is selecting the proper structure or "ansatz" for the efficient circuit regarding depth, width, and parameters while remaining robust and versatile. This challenge is similar to a fundamental problem in classical machine learning, which involves creating simple yet effective models. Nevertheless, the challenge continues beyond model selection. Training these quantum models differs from traditional approaches, as they are based on physical quantum algorithms rather than mathematical formulas. That raises questions about whether we can improve upon traditional numerical optimisation techniques. As a result, there are still some unanswered questions and uncertainties. In the context of quantum systems, choosing optimal parameterisation techniques is crucial to achieving high-quality results. It is imperative to consider whether classical iterative training strategies are adequate or if new techniques should be devised to suit the unique characteristics of quantum systems. As such, it needs to address whether existing parameterisation methods are fit for purpose or whether novel approaches need to be developed to meet the specific requirements of quantum systems. In essence, variational algorithms are not just another tool; they may hold the key to a whole new dimension in machine learning. They could pave the way for a broader paradigm in which we use physical devices as machine learning models, guided by our classical computers during training.

Transferring these investigations from quantum computer simulators to real quantum computers is desirable once the simple tests necessary to validate the model have been completed. Despite the excellent performance and optimisations gained so far of both hardware and software on the enormous amount of calculations to be performed, the simulators still need to be able to exploit the quantum peculiarities of matter, which allow parallelism in the calculation, which is not classically attainable.

Future investigations will focus on this front. That will provide insight into the problems arising from the physical implementation and the feasibility of the proposed solutions.

ACKNOWLEDGMENT

The authors thank Grammarly Inc., Quillbot Inc., Deepl SE, and Google LLC for the use of their platform in order to improve the concepts they wanted to formulate in English.

ABBREVIATIONS

The following abbreviations are used in this manuscript:

CV	Continuous Variables
FFNN	Feed-Forward Neural Network
NISQ	Noisy Intermediate-Scale Quantum
OQFFNN	Optical Quantum FFNN
QNN	Quantum Neural Network
ROC	Receiver Operating Characteristic
SGD	Stochastic Gradient Descent
SVD	Singular Value Decomposition
VQA	Variational Quantum Algorithms

AUTHOR CONTRIBUTIONS

Conceptualization: Marco Simonetti, Damiano Perri, and Osvaldo Gervasi; Software: Marco Simonetti; Data Curation: Marco Simonetti, Damiano Perri, and Osvaldo Gervasi; Investigation: Marco Simonetti, Damiano Perri, and Osvaldo Gervasi; Methodology: Marco Simonetti, Damiano Perri, and Osvaldo Gervasi; Supervision: Osvaldo Gervasi; Validation: Marco Simonetti, Damiano Perri, and Osvaldo Gervasi; Writing—Original Draft: Marco Simonetti; Writing—Review and Editing: Marco Simonetti, Damiano Perri, and Osvaldo Gervasi.

REFERENCES

[1] Z. Wu, Y. Tang, B. Hong, B. Liang, and Y. Liu, "Enhanced precision in dam crack width measurement: Leveraging advanced lightweight network identification for pixel-level accuracy," *Int. J. Intell. Syst.*, vol. 2023, pp. 1–16, Sep. 2023.

[2] Y. Tang, Y. Wang, D. Wu, M. Chen, L. Pang, J. Sun, W. Feng, and X. Wang, "Exploring temperature-resilient recycled aggregate concrete with waste rubber: An experimental and multi-objective optimization analysis," *Rev. Adv. Mater. Sci.*, vol. 62, no. 1, Aug. 2023, Art. no. 20230347.

[3] K. H. Wan, O. Dahlsten, H. Kristjánsson, R. Gardner, and M. S. Kim, "Quantum generalisation of feedforward neural networks," *Npj Quantum Inf.*, vol. 3, no. 1, pp. 1–8, Sep. 2017.

[4] M. Schuld and F. Petruccione, *Machine Learning With Quantum Computers*. Berlin, Germany: Springer, 2021.

[5] D. Deutsch and R. Jozsa, "Rapid solution of problems by quantum computation," *Proc. Roy. Soc. London. Ser. A, Math. Phys. Sci.*, vol. 439, no. 1907, pp. 553–558, Dec. 1992.

[6] L. K. Grover, "Quantum mechanics helps in searching for a needle in a haystack," *Phys. Rev. Lett.*, vol. 79, no. 2, p. 325, Jul. 1997.

[7] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Rev.*, vol. 41, no. 2, pp. 303–332, Jan. 1999.

[8] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, Aug. 2018.

[9] F. Leymann and J. Barzen, "The bitter truth about gate-based quantum algorithms in the NISQ era," *Quantum Sci. Technol.*, vol. 5, no. 4, Oct. 2020, Art. no. 044007.

[10] M. Salm, J. Barzen, F. Leymann, and B. Weder, "About a criterion of successfully executing a circuit in the NISQ era: What $wd \ll 1/\epsilon_{\mathrm{eff}}$ really means," in *Proc. 1st ACM SIGSOFT Int. Workshop Architectures Paradigms Eng. Quantum Softw.*, Nov. 2020, pp. 10–13.

[11] E. Zeidler, *Nonlinear Functional Analysis and Its Applications: III: Variational Methods and Optimization*. Berlin, Germany: Springer, 2013.

[12] D. R. Smith, *Variational Methods in Optimization*. Chelmsford, MA, USA: Courier Corporation, 1998.

[13] R. Borghi, "The variational method in quantum mechanics: An elementary introduction," *Eur. J. Phys.*, vol. 39, no. 3, May 2018, Art. no. 035410.

[14] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, and L. Cincio, "Variational quantum algorithms," *Nature Rev. Phys.*, vol. 3, no. 9, pp. 625–644, 2021.

[15] M. Weigold, J. Barzen, F. Leymann, and M. Salm, "Data encoding patterns for quantum computing," in *Proc. 27th Conf. Pattern Lang. Programs*, 2020, pp. 1–11.

[16] V. Vedral, A. Barenco, and A. Ekert, "Quantum networks for elementary arithmetic operations," *Phys. Rev. A, Gen. Phys.*, vol. 54, no. 1, pp. 147–153, Jul. 1996.

[17] J. A. Cortese and T. M. Braje, "Loading classical data into a quantum computer," 2018, *arXiv:1803.01958*.

[18] R. LaRose and B. Coyle, "Robust data encodings for quantum classifiers," *Phys. Rev. A, Gen. Phys.*, vol. 102, no. 3, Sep. 2020, Art. no. 032420.

[19] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum algorithm for linear systems of equations," *Phys. Rev. Lett.*, vol. 103, no. 15, Oct. 2009, Art. no. 150502.

[20] M. Schuld, M. Fingerhuth, and F. Petruccione, "Implementing a distance-based classifier with a quantum interference circuit," *Europhys. Lett.*, vol. 119, no. 6, p. 60002, Sep. 2017.

[21] M. Weigold, J. Barzen, F. Leymann, and M. Salm, "Expanding data encoding patterns for quantum algorithms," in *Proc. IEEE 18th Int. Conf. Softw. Archit. Companion (ICSA-C)*, Mar. 2021, pp. 95–101.

[22] E. Grant, M. Benedetti, S. Cao, A. Hallam, J. Lockhart, V. Stojevic, A. G. Green, and S. Severini, "Hierarchical quantum classifiers," *npj Quantum Inf.*, vol. 4, no. 1, pp. 1–8, Dec. 2018.

[23] N. Killoran, T. R. Bromley, J. M. Arrazola, M. Schuld, N. Quesada, and S. Lloyd, "Continuous-variable quantum neural networks," *Phys. Rev. Res.*, vol. 1, no. 3, Oct. 2019, Art. no. 033063.

[24] M. Simonetti, D. Perri, and O. Gervasi, "An example of use of variational methods in quantum machine learning," in *Proc. Int. Conf. Comput. Sci. Appl.* Switzerland: Springer, 2022, pp. 597–609.

[25] G. Biondi, V. Franzoni, O. Gervasi, and D. Perri, "An approach for improving automatic mouth emotion recognition," in *Proc. Int. Conf. Comput. Sci. Appl.* Switzerland: Springer, 2019, pp. 649–664.

[26] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum algorithms for supervised and unsupervised machine learning," 2013, *arXiv:1307.0411*.

[27] N. Wiebe, A. Kapoor, and K. Svore, "Quantum algorithms for nearest-neighbor methods for supervised and unsupervised learning," 2014, *arXiv:1401.2142*.

[28] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum principal component analysis," *Nature Phys.*, vol. 10, pp. 631–633, Jul. 2014.

[29] A. Lagana, O. Gervasi, S. Tasso, D. Perri, and F. Franciosa, "The ECTN virtual education community prosumer model for promoting and assessing chemical knowledge," in *Proc. Int. Conf. Comput. Sci. Appl.* Switzerland: Springer, 2018, pp. 533–548.

[30] F. Zollo, L. Caprini, O. Gervasi, and A. Costantini, "X3DMMS: An X3DOM tool for molecular and material sciences," in *Proc. 16th Int. Conf. 3D Web Technol.*, Jun. 2011, pp. 129–136.

[31] M. Mariotti, O. Gervasi, F. Vella, A. Cuzzocrea, and A. Costantini, ''Strategies and systems towards grids and clouds integration: A DBMS-based solution,'' *Future Gener. Comput. Syst.*, vol. 88, pp. 718–729, Nov. 2018.

[32] F. Vella, I. Neri, O. Gervasi, and S. Tasso, ''A simulation framework for scheduling performance evaluation on CPU-GPU heterogeneous system,'' in *Proc. Int. Conf. Comput. Sci. Appl.* Switzerland: Springer, 2012, pp. 457–469.

[33] M. Riganelli, V. Franzoni, O. Gervasi, and S. Tasso, ''EmEx, a tool for automated emotive face recognition using convolutional neural networks,'' in *Proc. Int. Conf. Comput. Sci. Appl.* Switzerland: Springer, 2017, pp. 692–704.

[34] F. Vella, R. M. Cefal, A. Costantini, O. Gervasi, and C. Tanci, ''GPU computing in EGI environment using a cloud approach,'' in *Proc. Int. Conf. Comput. Sci. Appl.*, Jun. 2011, pp. 150–155.

[35] D. Perri, P. S. Labini, O. Gervasi, S. Tasso, and F. Vella, ''Towards a learning-based performance modeling for accelerating deep neural networks,'' in *Proc. Int. Conf. Comput. Sci. Appl.* Switzerland: Springer, 2019, pp. 665–676.

[36] M. Simonetti, D. Perri, N. Amato, and O. Gervasi, ''Teaching math with the help of virtual reality,'' in *Proc. Int. Conf. Comput. Sci. Appl.* Switzerland: Springer, 2020, pp. 799–809.

[37] D. Perri, M. Simonetti, A. Lombardi, N. Faginas-Lago, and O. Gervasi, ''Binary classification of proteins by a machine learning approach,'' in *Proc. Int. Conf. Comput. Sci. Appl.* Switzerland: Springer, 2020, pp. 549–558.

[38] P. Benedetti, D. Perri, M. Simonetti, O. Gervasi, G. Reali, and M. Femminella, ''Skin cancer classification using inception network and transfer learning,'' in *Proc. Int. Conf. Comput. Sci. Appl.*, Switzerland, 2020, pp. 536–545.

[39] D. Perri, M. Simonetti, A. Lombardi, N. Faginas-Lago, and O. Gervasi, ''A new method for binary classification of proteins with machine learning,'' in *Proc. Int. Conf. Comput. Sci. Appl.* Switzerland: Springer, 2021, pp. 388–397.

[40] S. Yoo, J. Bang, C. Lee, and J. Lee, ''A quantum speedup in machine learning: Finding an N-bit Boolean function for a classification,'' *New J. Phys.*, vol. 16, no. 10, Oct. 2014, Art. no. 103014.

[41] Y. Cao, G. G. Guerreschi, and A. Aspuru-Guzik, ''Quantum neuron: An elementary building block for machine learning on quantum computers,'' 2017, *arXiv:1711.11240*.

[42] A. Narayanan and T. Menneer, ''Quantum artificial neural network architectures and components,'' *Inf. Sci.*, vol. 128, nos. 3–4, pp. 231–255, Oct. 2000.

[43] S. Mangini, F. Tacchino, D. Gerace, D. Bajoni, and C. Macchiavello, ''Quantum computing models for artificial neural networks,'' *Europhys. Lett.*, vol. 134, no. 1, p. 10002, Apr. 2021.

[44] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, ''The power of quantum neural networks,'' *Nature Comput. Sci.*, vol. 1, no. 6, pp. 403–409, 2021.

[45] C. Ciliberto, M. Herbster, A. D. Ialongo, M. Pontil, A. Rocchetto, S. Severini, and L. Wossnig, ''Quantum machine learning: A classical perspective,'' *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 474, no. 2209, 2018, Art. no. 20170551.

[46] V. Dunjko and H. J. Briegel, ''Machine learning & artificial intelligence in the quantum domain: A review of recent progress,'' *Rep. Prog. Phys.*, vol. 81, no. 7, Jul. 2018, Art. no. 074001.

[47] G. Adesso, S. Ragy, and A. R. Lee, ''Continuous variable quantum information: Gaussian states and beyond,'' *Open Syst. Inf. Dyn.*, vol. 21, Jun. 2014, Art. no. 1440001.

[48] A. Serafini, *Quantum Continuous Variables: A Primer of Theoretical Methods*. Boca Raton, FL, USA: CRC Press, 2017.

[49] A. Ferraro, S. Olivares, and M. G. A. Paris, ''Gaussian states in continuous variable quantum information,'' 2005, *arXiv:0503237*.

[50] C. Weedbrook, S. Pirandola, R. García-Patrón, N. J. Cerf, T. C. Ralph, J. H. Shapiro, and S. Lloyd, ''Gaussian quantum information,'' *Rev. Mod. Phys.*, vol. 84, p. 621, May 2012.

[51] N. Fabre, M. Amanti, F. Baboux, A. Keller, S. Ducci, and P. Milman, ''Correction: The Hong–Ou–Mandel experiment: From photon indistinguishability to continuous-variable quantum computing,'' *Eur. Phys. J. D*, vol. 76, no. 12, p. 196, Dec. 2022.

[52] T. Hillmann, F. Quijandría, G. Johansson, A. Ferraro, S. Gasparinetti, and G. Ferrini, ''Universal gate set for continuous-variable quantum computation with microwave circuits,'' *Phys. Rev. Lett.*, vol. 125, no. 16, Oct. 2020, Art. no. 160501.

[53] J. Nokkala, R. Martínez-Peña, G. L. Giorgi, V. Parigi, M. C. Soriano, and R. Zambrini, ''Gaussian states of continuous-variable quantum systems provide universal and versatile reservoir computing,'' *Commun. Phys.*, vol. 4, no. 1, p. 53, Mar. 2021.

[54] H. Yang, W. Roga, J. D. Pritchard, and J. Jeffers, ''Gaussian state-based quantum illumination with simple photodetection,'' *Opt. Exp.*, vol. 29, no. 6, pp. 8199–8215, 2021.

[55] M. Walschaers, ''Non-Gaussian quantum states and where to find them,'' *PRX Quantum*, vol. 2, no. 3, Sep. 2021, Art. no. 030204.

[56] E. Knill, R. Laflamme, and G. J. Milburn, ''A scheme for efficient quantum computation with linear optics,'' *Nature*, vol. 409, no. 6816, pp. 46–52, Jan. 2001.

[57] J. M. Arrazola, T. R. Bromley, J. Izaac, C. R. Myers, K. Brádler, and N. Killoran, ''Machine learning method for state preparation and gate synthesis on photonic quantum computers,'' *Quantum Sci. Technol.*, vol. 4, no. 2, Jan. 2019, Art. no. 024004.

[58] H.-K. Lau, R. Pooser, G. Siopsis, and C. Weedbrook, ''Quantum machine learning over infinite dimensions,'' *Phys. Rev. Lett.*, vol. 118, no. 8, Feb. 2017, Art. no. 080501.

[59] K. Bartkiewicz, C. Gneiting, A. Cernoch, K. Jiráková, K. Lemr, and F. Nori, ''Experimental kernel-based quantum machine learning in finite feature space,'' *Sci. Rep.*, vol. 10, no. 1, pp. 1–9, Jul. 2020.

[60] J. M. Arrazola, V. Bergholm, K. Bradler, T. R. Bromley, M. J. Collins, I. Dhand, A. Fumagalli, T. Gerrits, A. Goussev, and L. G. Helt, ''Quantum circuits with many photons on a programmable nanophotonic chip,'' *Nature*, vol. 591, no. 7848, pp. 54–60, Mar. 2021.

[61] R. Prevedel, P. Walther, F. Tiefenbacher, P. Böhi, R. Kaltenbaek, T. Jennewein, and A. Zeilinger, ''High-speed linear optics quantum computing using active feed-forward,'' *Nature*, vol. 445, no. 7123, pp. 65–69, Jan. 2007.

[62] N. K. Langford, S. Ramelow, R. Prevedel, W. J. Munro, G. J. Milburn, and A. Zeilinger, ''Efficient quantum computing using coherent photon conversion,'' *Nature*, vol. 478, no. 7369, pp. 360–363, Oct. 2011.

[63] A. A. Melnikov, H. P. Nautrup, M. Krenn, V. Dunjko, M. Tiersch, A. Zeilinger, and H. J. Briegel, ''Active learning machine learns to create new quantum experiments,'' *Proc. Nat. Acad. Sci. USA*, vol. 115, no. 6, pp. 1221–1226, Feb. 2018.

[64] A. Rocchetto, S. Aaronson, S. Severini, G. Carvacho, D. Poderini, I. Agresti, M. Bentivegna, and F. Sciarrino, ''Experimental learning of quantum states,'' *Sci. Adv.*, vol. 5, no. 3, Mar. 2019, Art. no. eaau1946.

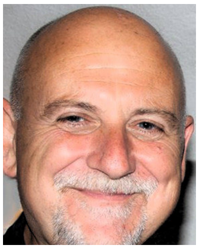
[65] V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, M. S. Alam, S. Ahmed, J. M. Arrazola, C. Blank, A. Delgado, and S. Jahangiri, ''PennyLane: Automatic differentiation of hybrid quantum-classical computations,'' 2018, *arXiv:1811.04968*.

[66] N. Killoran, J. Izaac, N. Quesada, V. Bergholm, M. Amy, and C. Weedbrook, ''Strawberry fields: A software platform for photonic quantum computing,'' *Quantum*, vol. 3, p. 129, Mar. 2019.

[67] M. Schuld and N. Killoran, ''Is quantum advantage the right goal for quantum machine learning?'' *PRX Quantum*, vol. 3, no. 3, Jul. 2022, Art. no. 030101.

MARCO SIMONETTI graduated in electronic engineering from the University of Bologna, with a thesis on chaotic carrier transmission systems. He is currently pursuing the joint Ph.D. degree in computer science with the Program of Mathematics, Statistics, and Computer Science, Departments of Mathematics and Computer Science, University of Florence, and the University of Perugia. He worked in the field of clinical engineering for ten years and participated with several teams in the implementation of various automated decision-making systems for health administrations. He is a High School Mathematics Teacher. His current research interests include deep learning and neural networks, quantum computing, and mathematics education.

DAMIANO PERRI received the B.Sc. and M.Sc. (summa cum laude) degrees in programming and management of computer systems from Perugia University and the joint Ph.D. degree in computer science with the Program of Mathematics, Statistics and Computer Science, Departments of Mathematics and Computer Science, University of Florence, and the University of Perugia. He teaches elements of computer architecture and operating systems. He is currently a Research Fellow with the Department of Mathematics and Computer Science, University of Perugia. His current research interests include virtual and augmented reality, cloud computing, machine learning, GPGPU computing, parallel and distributed environments, and neural networks. In particular, his recent research activity is aimed at investigating predictive models to optimize the most important operators in emerging convolutional neural networks (CNNs). He is an expert developer of synthetic environments and their applications in everyday life. Since 2015, he has been the Technical Manager of the LibreEOL Project (https://www.libreeol.org/info). He has been a University Tutor for various courses, including virtual reality, high-performance computing, and programming with Python. He is an ACM Member. He served in the important role of the Technology Chair for the 2020, 2021, 2022, and 2023 Edition of the ICCSA Conference (https://www.iccsa.org).



OSVALDO GERVASI (Senior Member, IEEE) received the bachelor's degree in computer science (networks architecture and virtual reality systems) and the master's degree in computer science (high-performance computing and human–computer interaction) from Perugia University. He has been the Deputy Director of the Department of Mathematics and Computer Science, since 2019. He is currently an Associate Professor with the Department of Mathematics and Computer Science, University of Perugia. He is also the Director of the professional degree course in computer systems programming and management. He has published over 150 papers in scientific journals and peer-reviewed international conference proceedings. He edited more than 100 books of international conference proceedings. His current research interests include parallel and distributed computing, cloud and high-performance computing, GPGPU computing, artificial intelligence, virtual and augmented reality, and e-learning. He is a Senior Member of ACM and a member of the Internet Society. He was invited seven times to deliver a plenary lecture at international conferences. He was the President of the Open-Source Competence Centre (CCOS), Umbria Region, Italy, established under the Regional Law 11/2006, from 2007 to 2013. He is the President of the not-for-profit organization ICCSA, which manages the International Conference on Computational Science and Its Applications (https://www.iccsa.org). He served on the board of directors of The Document Foundation (TDF), which released the popular open-source Office Suite, LibreOffice, from 2016 to 2020. He is the Project Leader of the Open-Source E-Assessment Platform LibreEOL (https://www.libreeol.org/info).

Open Access funding provided by 'Università degli Studi di Perugia' within the CRUI CARE Agreement