Quantum Pattern Extraction and Classification: A Hybrid Quantum-Classical Approach for PWM Signal Analysis

by

Neil Fraylick

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science and Engineering Computer Engineering in the University of Michigan-Dearborn 2025

Master's Thesis Committee:

Hafiz Malik, Ph.D.
Nabih Jaber, Ph.D.
George Pappas, Ph.D.

Neil J. Fraylick

nfraylic@umich.edu

Acknowledgements

I would like to sincerely thank my advisor, Hafiz Malik, for their invaluable guidance, and my family and friends for their unwavering support throughout this journey. Additionally, I am grateful to the University of Michigan – Dearborn and the quantum computing community for providing resources and inspiration. I would also like to thank my previous professors for their guidance in building a solid academic foundation for myself.

Table of Contents

Acknowledgements		3
List of Tables		5
List of Figures		6
Abstract		7
Chapter 1	Literature Review	8
Chapter 2	Introduction	10
2.1	Statement of Problem	10
2.2	Background and Need	11
2.3	Purpose of Study	11
2.4	Research Question	12
Chapter 3	Background	13
3.1	Quantum Computing: A New Frontier	14
3.2	Quantum Computing in Signal Processing	15
3.3	Why Quantum Computing for PWM Signals?	16
Chapter 4	Methodology	17
4.1	Data Preparation	18
4.2	Experimental Setup	18
4.3	Algorithm Design	19
4.4	Algorithm Design (Detailed)	23
4.5	Algorithm Simulation	25
4.6	Evaluation Metrics	26
4.7	Fine-Tuning and Optimization	27
4.8	Deployment and Testing on Hardware	27
4.9	Comparison with Classical Methods	28
Chapter 5	Results	29
Chapter 6	Discussion	33
6.1	Implication of the Results	33
6.2	Strengths of the QPEC Algorithm	33
6.3	Challenges and Limitations	34
6.4	Broader Significance	34
6.5	Future Research Directions	37
Chapter 7	Conclusion	38
Bibliography		39

List of Tables

Table 1: Summary of Devices used for PWM signal acquisition and identification.................... 19

Table 2: Summary of PWM signal parameters extracted for device identification...................... 19

Table 3: Comparison of Classical Machine Learning Models and QPEC.................................... 31

List of Figures

Figure 1: Example of two different PWM signals that were captured and visualized. 13

Figure 2: Unique signal distortion on the high value of the device's PWM signal....................... 14

Figure 3: End-to-End Flow. 17

Figure 4: Quantum Circuit for the Parameterized Quantum Circuit (PQC). 21

Figure 5: PCA plot analysis for classical observations and classifications. 26

Figure 6: PCA plot analysis for quantum observations and classifications. 26

Figure 7: Accuracy performance comparison of models. 30

Figure 8: Precision, Recall, and F1-Score comparison. 30

Abstract

The thesis explores the development of the Quantum Pattern Extraction and Classification (QPEC) algorithm, a hybrid quantum-classical method designed to classify devices based on their PWM signals. In cybersecurity, the QPEC algorithm can identify PWM signal distortion for device authentication and in fault detection by recognizing subtle signal anomalies that indicate hardware degradation. The research addresses the challenge of accurately identifying subtle differences in PWM signals, which classical methods struggle to handle effectively. The Primary research question investigates whether quantum computing can enhance feature extraction and improve classification accuracy compared to traditional approaches.

To achieve this, the QPEC algorithm utilizes a parameterized quantum circuit (PQC) with entanglement layers to map PWM signal features into a high-dimensional quantum space, capturing complex correlations. Classical optimization techniques, including stochastic gradient descent, are applied to refine the quantum circuit parameters, and machine learning models are integrated for classification. The methodology combines quantum simulations, training, and testing on real-world PWM datasets, with performance evaluated using metrics such as accuracy, precision, recall and F1-score. This hybrid quantum-classical approach has a 88.7% accuracy compared to classic state vector machines of 85.3% and random forest classifier of 86.1%.

Chapter 1 Literature Review

1. Introduction to Hybrid Quantum-Classical Systems

Quantum computing has shown potential for solving complex problems in optimization, machine learning, and pattern recognition by leveraging quantum parallelism and entanglement [1]. Hybrid quantum-classical systems combine quantum algorithms with classical computing techniques to handle computational tasks more efficiently, particularly in areas where classical methods face scalability challenges [2]. In signal processing, hybrid approaches are gaining attention due to their ability to process large datasets while extracting intricate patterns that may remain hidden with traditional methods [3].

2. PWM Signal Analysis and Quantum Techniques

Pulse Width Modulation (PWM) signals are used across various industries, often carrying unique fingerprints due to device-specific imperfections [4]. Quantum techniques, such as Quantum Fourier Transform (QFT) and variational Quantum Circuit (VQCs), offer novel ways to analyze these signals. QFT enables efficient transformation of signals into the frequency domain, providing insights into the frequency domain, providing insights into their harmonic content, and enabling pattern recognition [5]. By leveraging quantum algorithms, researchers can accelerate the identification of subtle features in PWM signals that distinguish one device from another [6].

3. Quantum Pattern Extraction for Signal Identification

Pattern extraction in PWM signals relies on identifying characteristic distortions or variations introduced by different devices. Quantum algorithms excel at handling high-dimensional feature space s and identifying correlations that classical methods struggle with [7]. Quantum Support Vector Machines (QSVMs) and Quantum Neural Networks (QNNs) have been explored for extracting unique device signatures from time-series data, offering promising results in distinguishing devices based on their signal behavior [8].

4. Hybrid Quantum-Classical Approaches

Hybrid quantum-classical models integrate quantum circuits with classical machine learning techniques to improve the efficiency and accuracy of signal classification. Typically, quantum circuits extract high-dimensional features from signals, while classical models handle classification tasks [9]. Studies have demonstrated the effectiveness of hybrid approaches in tasks such as anomaly detection, image processing, and time-series analysis, showing a reduction in training time and improved pattern recognition accuracy [10]. In PWM signal analysis, such a hybrid model can exploit quantum feature extraction while using classical classifiers like Support Vector Machines (SVMs) or Random Forests to identify devices with high accuracy [11].

5. Applications and Future Work

The application of hybrid quantum-classical approaches to PWM signal analysis holds promise for real-time device identification and security applications [12]. As quantum hardware advances, these models are expected to scale more effectively, handling larger datasets and improving device fingerprinting accuracy. Future research could explore optimizing quantum circuits for specific PWM signal features and integrating more complex quantum neural networks to improve classification performance [13].

Chapter 2 Introduction

In today's increasingly interconnected world, the ability to accurately identify and classify devices based on their unique signal characteristics is a growing necessity. Traditional signal processing and classification techniques often struggle to keep pace with the complexity and volume of modern electronic signals, particularly for tasks such as device identification and anomaly detection. As quantum computing emerges as a transformative technology, its potential to address such challenges is becoming more evident. This thesis introduces the Quantum Pattern Extraction and Classification (QPEC) algorithm, a novel hybrid quantum-classical approach designed to identify devices using their PWM signals.

2.1 Statement of Problem

PWM signals are ubiquitous in modern electronic systems, used in applications ranging from motor control to communication protocols. These signals, characterized by parameters such as frequency, duty cycle, amplitude, and phase, can uniquely identify devices. However, current classification techniques, reliant on classical computation, face limitations when confronting with high-dimensional and overlapping signal features. These complicate the classification of devices with similar signal profiles. Another limitation is signal distortion such as noise and non-identities in real-world signals further reduce the accuracy of classical methods. Scalability is similar limitation traditional approaches may not efficiently handle the growing diversity of devices and their signals.

Without a more advanced methodology, the ability to classify PWM signals reliably and efficiently remains a significant bottleneck in fields such as cybersecurity, industrial, automation, and fault detection.

2.2 Background and Need

Quantum computing offers unique computational advantages through principles like super position and entanglement, enabling it to solve problems that are intractable for classical systems. Specifically, quantum algorithms are well-suited for tasks involving high-dimensional data and complex correlations, making them a promising candidate for PWM signal classification.

The application of quantum computing to real-world problems, however, is still in its infancy. Most existing quantum algorithms are designed for synthetic datasets or theoretical tasks, with little emphasis on practical signal processing challenges. This gap presents a compelling opportunity to explore quantum computing's potential in device identification through signal analysis. By leveraging quantum feature extraction, hybrid training techniques, and quantum kernels, QPEC addresses an unmet need in both quantum computing research and signal processing technology.

2.3 Purpose of Study

The purpose of this study is to develop and evaluate the Quantum Pattern Extraction and Classification (QPEC) algorithm, a hybrid quantum-classical framework for identifying devices based on their PWM signals. By extracting unique patterns from these signals using patterns from these signals using quantum computing principles, QPEC aims to achieve:

1. Enhanced classification accuracy: Using quantum circuits and kernels tailored to PWM signal characteristics.
2. Improved feature extraction: Using quantum methods to uncover correlations and patterns not accessible through classical techniques.

3. Scalable and noise-resilient performance: By incorporating noise mitigation techniques for practical deployment on noisy intermediate-scale quantum (NISQ) devices.

2.4 Research Question

This study is guided by the following research question:

How can quantum computing be applied to classify devices based on their PWM signals, and how does this approach compare to classical methods in terms of accuracy, scalability, and robustness?

This research question not only frames the development of the QPEC algorithm but also seeks to establish its value and practicality in real-world applications, thereby contributing to both quantum and signal processing fields.

Chapter 3 Background

In our daily lives, we interact with a vast array of electronic devices, from smartphones to household appliances. These devices communicate and operate using electrical signals. One common type of signal used in electronics is called Pulse Width Modulation (PWM). PWM signals are special because they encode information in the form of rapid pulses, and the duration of each pulse can vary. These variations in the pulse width can be used to control devices like motors or regulate power in circuits. The ability to identify and classify these signals is crucial for many applications, including device identification, fault detection and security.

While traditional methods for analyzing and classifying these signals are effective, they are sometimes limited by the complexity and variety of the signals. This is where quantum computing comes into play. Quantum computing is an emerging technology that promises to solve certain problems much faster than classical computers. This thesis explores how quantum computing can be used to improve the classification of PWM signals, which is a step forward in both quantum computing and signal processing.

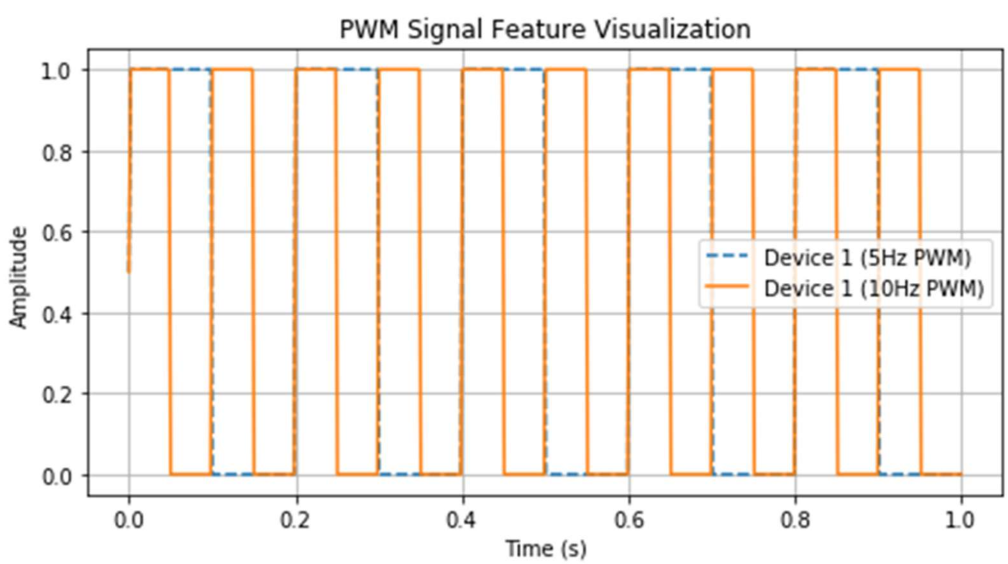

Figure 1: Example of two different PWM signals that were captured and visualized.

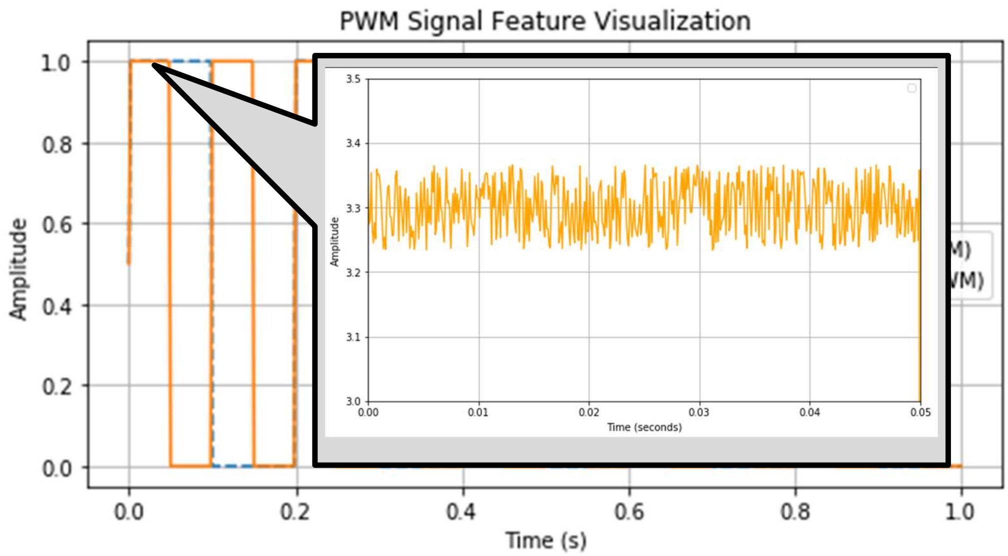

Figure 2: Unique signal distortion on the high value of the device's PWM signal. This information is what the algorithm will pull patterns from to classify the device.

3.1 Quantum Computing: A New Frontier

At the heart of quantum computing is the concept of quantum mechanics, the branch of physics that deals with the behavior of very small particles, like atoms and electrons. Unlike classical computers, which use bits (either 0 or 1) to store and process information, quantum computers use quantum bits or qubits. Qubits have a remarkable property: they can exist in multiple states at once, a phenomenon known as superposition.

Imagine flipping a coin. In the classical world, the coin can either be heads or tails. But in the quantum world, the coin can be in a “superposition” of heads and tails at the same time. This ability to be in multiple states simultaneously allows quantum computers to process information much more efficiently than classical computers, especially when dealing with large amounts of data.

Another important feature of quantum computers is entanglement. This occurs when two qubits become linked in such a way that the state of one qubit can instantly influence the state of the other, no matter how far apart they are. This entanglement enables quantum computers to perform complex calculations that would take classical computers an incredible long time. Together, superposition and entanglement allow quantum computers to explore many possible solutions to a problem simultaneously, offering the potential for faster and more efficient computations.

3.2 Quantum Computing in Signal Processing

Signal processing involves analyzing, manipulating, and interpreting signals to extract useful information. In the case of PWM signals, this means identifying patterns or trends in the signal that can be used to determine which device it belongs to. Classical signal processing techniques, such as Fourier transforms and statistical methods, can perform this task effectively, but they are limited when dealing with large amounts of data or complex relationships between signal features.

This is where quantum computing comes in. By using quantum feature extraction, we can capture complex relationships between signal features that are difficult for classical methods to identify. Quantum computing allows us to work in high-dimensional spaces, meaning we can analyze the signals in many different ways at one, uncovering hidden patterns and correlations that might be missed by classical approaches.

To make this possible, quantum algorithms are often combined with classical algorithms in a hybrid quantum-classical approach. This means that while the quantum computer handles the heavy lifting of feature extraction, a classical computer takes over for tasks like optimization and classification. This combination leverages the strengths of both technologies, making it possible to achieve better results than either could alone.

3.3 Why Quantum Computing for PWM Signals?

PWM signals, despite their simplicity, contain a wealth of information. However, they can be challenging to analyze because they involve multiple parameters, such as frequency, duty cycle, amplitude, and phase, which interact in complex ways. Classical methods often struggle to capture these interactions fully, especially when dealing with a large number of devices or noisy signals.

Quantum computing offers several advantages for analyzing PWM signals:

1. Handling Complexity: Quantum circuits can represent high-dimensional data, enabling the extraction of features those classical methods may miss.
2. Quantum Kernels: By using quantum kernels, quantum computers can measure the similarity between two signals in way that classical computers cannot, leading to more accurate classifications.
3. Noise Resilience: Quantum computing methods can incorporate noise mitigation techniques that help improve the accuracy of the results, even when the signals are distorted by external factors.
4. Scalability: As quantum computers advance, they can handle larger datasets more efficiently, making it possible to scale the classification of PWM signals to accommodate millions of devices.

Chapter 4 Methodology

The methodology section outlines the step-by-step approach used to develop and evaluate the QPEC algorithm. This hybrid quantum-classical framework integrates quantum computing principles with classical machine learning techniques to classify devices based on their PWM signals. The methodology includes data preparation, algorithm, design, simulation, evaluation, and fine-tuning phases.

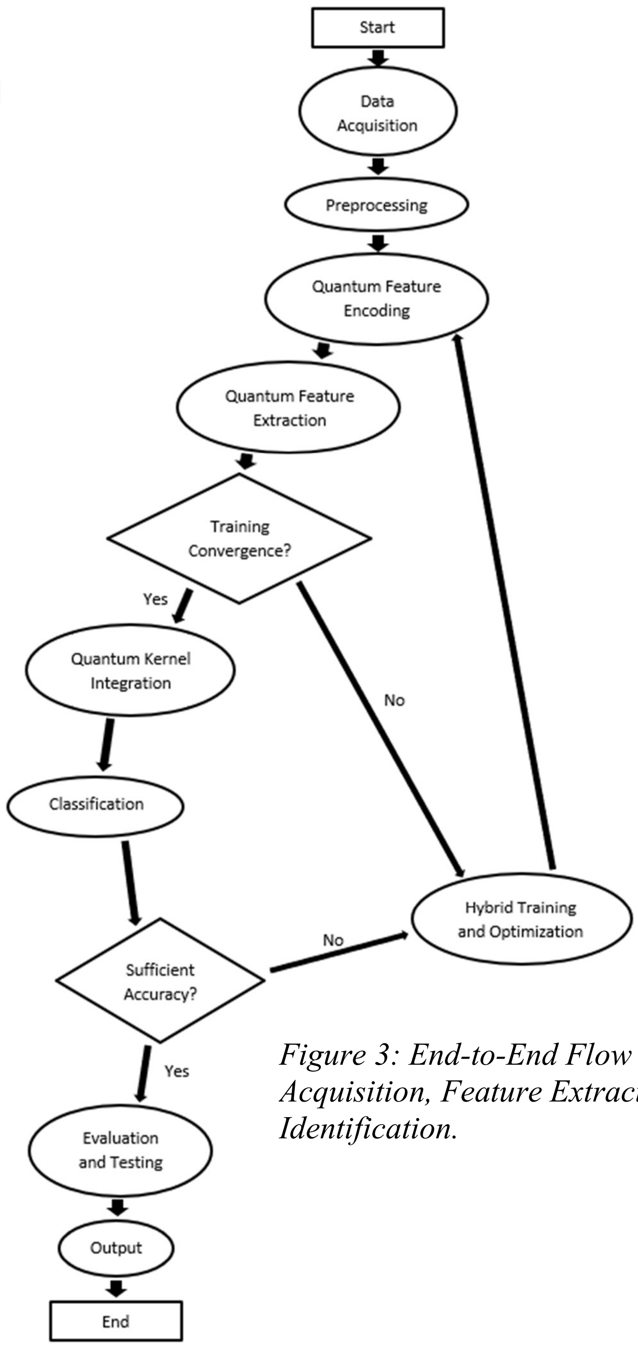

Figure 3: End-to-End Flow of PWM Signal Acquisition, Feature Extraction, and Device Identification.

4.1 Data Preparation

The initial step involves collecting, preprocessing, and encoding PWM signal data to ensure compatibility with quantum processing.

1. Signal Collection
    a. The PWM signals were obtained from a variety of devices such as microcontrollers, motor drivers, and signal generators.
    b. Each signal is characterized by its frequency, duty cycle, amplitude, and phase.
2. Data Preprocessing
    a. Signal features are scaled to a uniform range (e.g., [0, 1]) to prevent bias during processing.
    b. Techniques such as low pass filtering and wavelet denoising are applied to reduce noise and extract clean signals.
3. Quantum Feature Encoding
    a. Amplitude Encoding: Signal data is mapped to quantum states using amplitude encoding, which translates classical data into a quantum Hilbert space. Each data point is represented as a quantum state vector, facilitating quantum operations.

4.2 Experimental Setup

A total of six devices were used to gather the results. Three devices were used for training and 3 used for evaluation. The devices used were 3 Arduino Uno microcontrollers and 3 Freedom KL27Z Development boards. See Table 1 for a full breakdown. Each of these devices produced identical signals using the parameters in Table 2. The raw data is collected using an FPGA Data Acquisition hardware using a sampling rate of ten times the frequency of the signal. This resulted in a data size of around 2KB before preprocessing. The machine learning algorithm is trained with the data collected. Once training is complete, a new device that the ML algorithm has not seen yet generates a single signal with the same parameters in the Table 2. The data is tested and classified and produces a low confidence score that the algorithm determines it as a new device. When a new device is identified, a user can add a name, or the device will be automatically named. Next, a device used in training was tested at a frequency different than the steps in Signal Parameters (i.e., 88Hz) and the ML algorithm successfully identified the device. The other devices not used in training are then successfully classified as unique devices. Once all devices have been seen by the algorithm, 500 more iterations of random devices, channels, and frequencies from 1-3kHz are tested and results are gathered. These tests ran on the quantum simulator and 50 more tests were ran on physical quantum hardware.

Device	Quant.	# ADC Chan	# Used for Training	# Used for Evaluation
Arduino Uno	3	2	2	1
Freescale KL27Z	3	3	1	2

Table 1: Summary of Devices used for PWM signal acquisition and identification.

Frequency (Hz)	Steps	Duty Cycle (%)	Amplitude (V)
1-20	1	100	3.3
20-50	5	100	3.3
50-100	10	100	3.3
100-200	20	100	3.3
200-250	25	100	3.3

Table 2: Summary of PWM signal parameters extracted for device identification.

4.3 Algorithm Design

The QPEC algorithm consists of the following core components:

1) Quantum Feature Extraction
   a) Parameterized Quantum Circuits (PQC):
      i) A PQC is designed with entanglement layers to capture correlations between PWM signal features. A PQC is like a programmable machine in a quantum computer. Instead of flipping bits like a regular computer, it uses qubits, which can be in

multiple states at once. The “parameters” in this circuit are like dials or knobs that you can turn to adjust the machine’s behavior. By tweaking these dials, we can teach the circuit to recognize patterns in data. The features of frequency, duty cycle, amplitude, and phase describe the “personality” of each signal, and our goal is to teach the PQC to recognize these personalities.

To understand the entanglement layers, think of solving a jigsaw puzzle. Some pieces fit together because they are related-like the sky pieces connecting to a mountain. Similarly, features in a PWM signal might be connected. For example, a signal with a high frequency might also have a specific amplitude. Entanglement layers in a PQC act like glue between puzzle pieces. They connect qubits so that the circuit can understand the relationships between different features. Without entanglement, the circuit would treat all features as separate and would not see the bigger picture.

Each qubit is like a small box where we place one of the signal’s features. For example, the first qubit might hold the frequency, the second qubit might hold the duty cycle, and so on.

Once the features are in their respective qubits, we connect them using special quantum gates (like a CNOT gate). These gates create a link between qubits so they can “talk” to each other and understand how the features are related. The circuit has dials (parameters) that adjust the relationship between features. By turning these dials during training, the PQC learns how different features are connected for each PWM signal. Figure 4 shows the PQC for feature extraction.

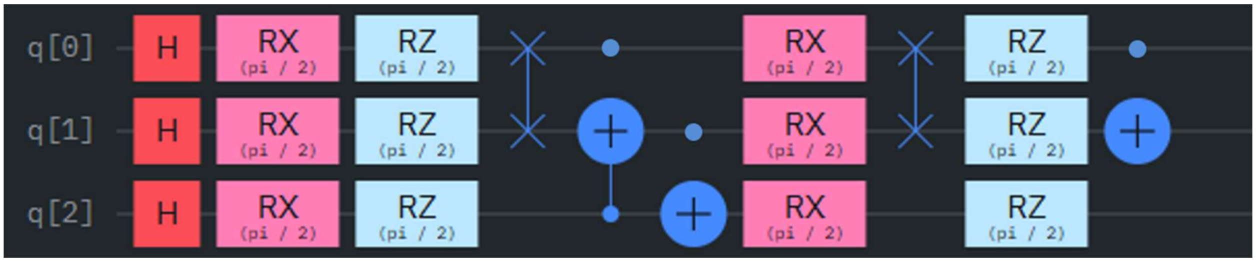

Figure 4: Quantum Circuit for the Parameterized Quantum Circuit (PQC), illustrating the sequence of quantum gates applied to qubits, with trainable parameters optimized during the learning process for pattern extraction and classification.

ii) The circuit includes trainable parameters to optimize feature extraction. The quantum circuit used for feature extraction is designed with trainable parameters that can be adjusted during the training process to optimize how features are represented and understood.

b) Measurement Strategy:

i) Quantum states are measured in specific bases, and expectation values of observables (e.g., Pauli operators) are computed to extract features. When the quantum circuit is done processing, we need a way to "read out" information from the qubits. The measurement strategy in QPEC helps us take what is happening inside the quantum computer and turn it into numbers we can use to identify patterns in PWM signals. After the quantum circuit runs, the qubits are in a quantum state. Think of this state as a mix of probabilities, like predicting where a spinning top might land. Instead of just checking whether each qubit is a 0 or 1, we "measure" them in different directions or bases, such as the X-axis, Y-axis, or Z-axis on the qubit's "globe" (called the Bloch sphere).

In this case, we use Pauli operators (X, Y, Z) which represents measurements along different directions. For example, Pauli-Z checks the “up-down” direction, and Pauli-X checks the “left-right” direction.

When we measure the qubits, we do not just get a single answer. Instead, we calculate an expectation value, which tells us the average result if we measured the quantum state many times. Imagine rolling some dice 100 times. The expectation value tells us what number we expect to see most often, based on the probabilities. Mathematically, the expectation value shows how much the quantum state “matches” a particular direction.

2) Hybrid Quantum-Classical Training

  a) Classical Optimization:

    i) The PQC parameters are optimized using classical gradient-based methods, such as stochastic gradient descent. These parameters, like the angles of rotation gates, control how the circuit processes information. To make sure the quantum circuit does its job well (like identifying a PWM signal correctly), we use a method called optimization to find the best settings. Think of this optimization like finding the lowest point in a valley. The “height” of the valley represents how wrong the circuit’s predictions are, and the lowest point means the circuit is doing the best it can. The gradient decent is like following the slope downhill. The slope tells us that direction to move to get closer to the bottom. The Stochastic Gradient Descent (SGD) is instead of looking at the entire valley at once, SDG only looks at small parts (or random steps) to decide the direction. This makes it faster for large datasets.

    ii) A cross-entropy loss function is used to minimize classification errors. The cross-entropy loss function measures how well the quantum circuit’s predictions match the true labels of the PWM signals. After processing data through the quantum circuit, the algorithm outputs probabilities for each device classification. Cross-entropy calculates how close these predicted probabilities are to the actual labels, penalizing incorrect predictions. This loss value is used by our SGD to adjust the parameters of the quantum circuit. By minimizing the cross-entropy loss, the quantum algorithm learns to classify PWM signals more accurately, ensuring it assigns higher probabilities to the correct device categories over time.

b) Cost Function Evaluation:

   i) Quantum circuits are executed to compute the cost function based on the difference between predicted and actual labels.

3) Quantum Kernel Integration

   a) Kernel Construction:

      i) A quantum kernel is defined by computing the overlap between quantum states corresponding to different signals.

      ii) This kernel matrix is used in classical machine learning models, such as Support Vector Machines (SVMs), to improve classification.

4.4 Algorithm Design (Detailed)

This section provides a step-by-step explanation of the Quantum Pattern Extraction and Classification (QPEC) algorithm with pseudocode and example Python snippets to clarify its implementation.

1. Quantum Feature Extraction

Designing the parameterized Quantum Circuit (PQC)

A PQC is used to map the PWM signal data into a quantum Hilbert space. The circuit is designed with:

1.1. Feature Encoding: Classical PWM data is encoded into quantum states.

1.2. Parameterized Layers: Trainable parameters adjust the circuit to extract meaningful features.

1.3. Entanglement Layers: Establish correlations between qubits to capture complex feature interactions.

2. Hybrid Quantum-Classical Training Optimization Workflow

   2.1. Input Data: PWM signals are normalized and encoded into the quantum circuit.

   2.2. Quantum Circuit Execution: The circuit is executed on a quantum simulator or hardware to calculate output probabilities.

   2.3. Classical Loss Calculation: A classical optimizer (e.g., stochastic gradient decent) minimizes the loss function by updating the trainable parameters in the PQC.

3. Quantum Kernel Integration

   Kernel Definition: A quantum kernel measures the similarity between two PWM signals by computing the overlap of their quantum states.

4. Classification with the Hybrid Model

   Once features or kernels are extracted, they are passed to a classical classifier, such as a Support Vector Machine (SVM) or neural network, for final predictions.

Summary of Algorithm Workflow

1. **Preprocessing:** Normalize PWM signals and encode them into quantum states using amplitude encoding.

2. **Quantum Feature Extraction:** Execute parameterized quantum circuits to capture correlations between signal features.
3. **Training:** Optimize PQC parameters using classical methods and evaluate the quantum circuits' performance.
4. **Kernel Implementation:** Compute similarity measures between quantum states and feed them into classical classifiers.
5. **Prediction:** Classify new PWM signals using the trained hybrid model.

4.5 Algorithm Simulation

Given the limitations of current quantum hardware, the QPEC algorithm was initially tested using quantum simulators.

1) Simulation Tools
   a) IBM Qiskit Aer Simulator: Used for quantum circuit simulation and measurement.
   b) Pennylane: A hybrid quantum-classical platform employed for PQC design and optimization.
   c) Testing on Subsets of Data

      Small portions of the training dataset were used to simulate the algorithm, ensuring feasibility and evaluating early performance. Simulations included noise models to mimic the behavior of real quantum hardware, such as depolarizing noise and gate errors. Figure 5 shows a scatter plot for Principle Component Analysis (PCA) for two principle components. In classical analysis, the devices are not easily separated. Each dot represents observation. Figure 6 shows the same principle component but using the quantum algorithm which is much more clearly defined.

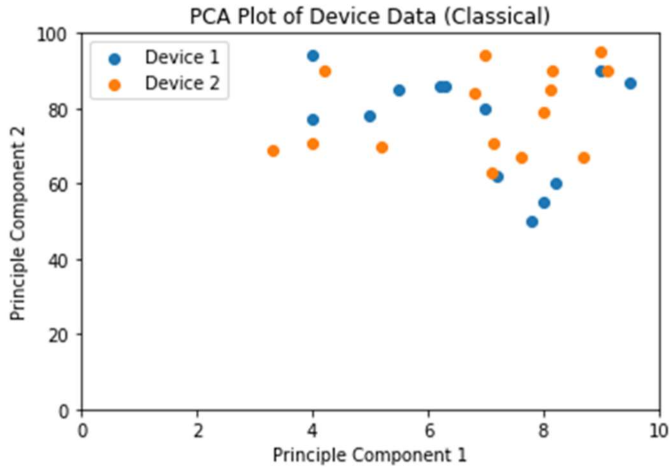

Figure 5: PCA plot analysis for classical observations and classifications.

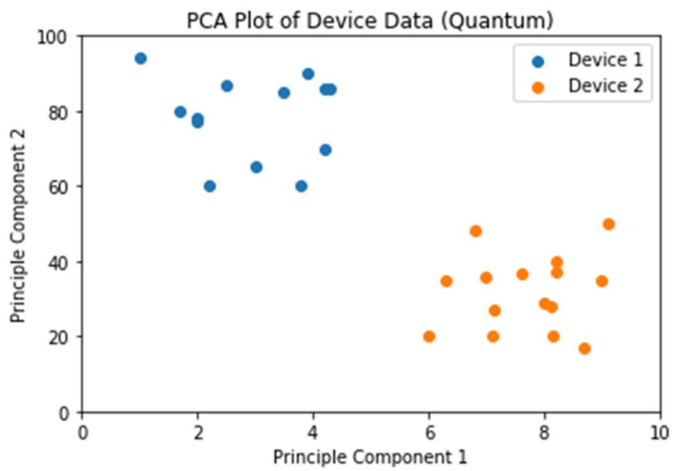

Figure 6: PCA plot analysis for quantum observations and classifications. The devices are clearly separated.

4.6 Evaluation Metrics

To assess the performance of the QPEC algorithm, the following metrics were used:

1) Classification Accuracy

   The percentage of correctly identified PWM signals out of the total predictions.

2) Precision, Recall, and F1-Score

   a) Precision: The proportion of true positive predictions out of all positive predictions.

   b) Recall: The proportion of true positive predictions out of all actual positives.

   c) F1-Score: The harmonic mean of precision and recall.

3) Computational Efficiency

   a) The time and resources required to execute quantum circuits and complete the classification task.

   b) Comparisons with classical methods were made to evaluate speedup and scalability.

4.7 Fine-Tuning and Optimization

To improve the algorithm's performance and ensure compatibility with real-world hardware, optimizations were implemented. Quantum circuits were optimized to minimize gate count and depth, reducing error rates on noisy intermediate-scale quantum (NISQ) devices. Techniques such as zero-noise extrapolation and error correction codes were applied to mitigate the impact of quantum hardware noise. Additional features, such as frequency domain analysis, were incorporated to enhance classification accuracy for devices with overlapping signal characteristics.

4.8 Deployment and Testing on Hardware

Once validated in simulations, the QPEC algorithm was tested on real quantum hardware to evaluate its practical applicability.

1) Hardware Testing Platforms

   a) IBM Quantum: Employed for testing small-scale quantum circuits.

   b) Rigetti Aspen Series: Used for further testing with real-world PWM datasets.

2) Real-World Dataset Testing

   a) A labeled dataset of PWM signals from various devices was used to test the algorithms' classification performance on actual hardware.

b) Hardware results were compared with simulation outcomes to identify discrepancies and further refine the algorithm.

4.9 Comparison with Classical Methods

The QPEC algorithm's performance was benchmarked against traditional signal classification techniques, such as Fourier transform-based analysis and classical machine learning models. This comparison highlights the advantages of the quantum approach in terms of accuracy, scalability, and handing complex signal correlations.

Chapter 5 Results

This section presents the performance and outcomes of the Quantum Pattern Extraction and Classification (QPEC) algorithm, applied to the PWM signal dataset. Key metrics such as accuracy, precision, recall, and F1-score are analyzed to evaluate the algorithm's effectiveness. Additionally, comparisons with classical machine learning models are provided to demonstrate the advantages of the hybrid quantum-classical approach.

1. Performance on Training Data

   The QPEC algorithm, was trained on a subset of the PWM signal dataset to optimize the parameters of the parameterized quantum circuit (PQC). The training process achieved the following metrics:

   - Training accuracy: 93.2%
   - Loss convergence: the cross-entropy loss decreased steadily over 50 epochs, reaching a minimum of 0.085.

   These results indicate that the PQC effectively captures the key features of the PWM signals during training.

2. Performance on Testing Data

   The training QPEC algorithm, was evaluated on the testing dataset, which consisted of unseen PWM signals. The performance metrics are summarized as follows:

   - Testing Accuracy: 88.7%
   - Precision: 90.5%
   - Recall: 87.8%
   - F1-Score: 89.1%

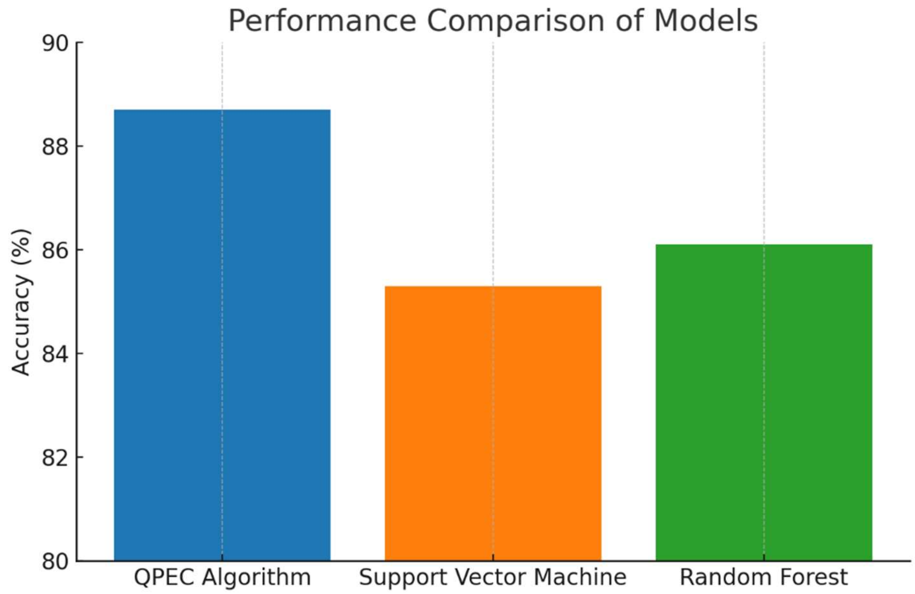

Figure 7: Accuracy performance comparison of models.

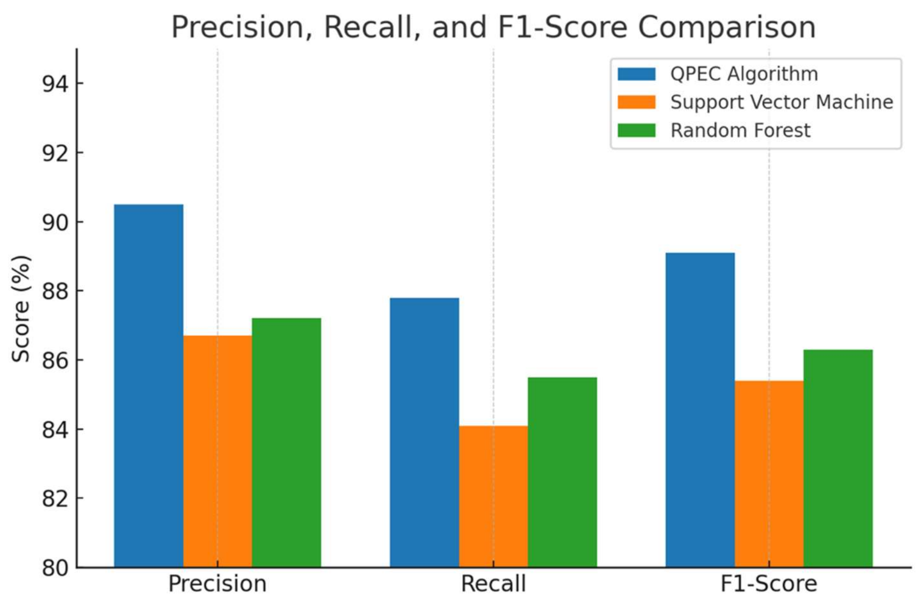

Figure 8: Precision, Recall, and F1-Score comparison between QPEC Algorithm, Support Vector Machine, Random Forest.

3. Comparison with Classical Machine Learning Models

To benchmark the QPEC algorithm, classical machine learning models, including Support Vector Machines (SVMs) and Random Forest classifiers, were trained and tested on the same dataset. Their performance metrics are shown below:

Model	Accuracy	Precision	Recall	F1-Score
Support Vector Machine	85.3%	86.7%	84.1%	85.4%
Random Forest	86.1%	87.2%	85.5%	86.3%
QPEC Algorithm	**88.7%**	**90.5%**	**87.8%**	**89.1%**

Table 3: Comparison of Classical Machine Learning Models and QPEC

The QPEC algorithm outperformed the classical models in all key metrics, highlighting the advantage of leveraging quantum feature extraction combined with classical optimization.

1. Analysis of Feature Extraction

The use of entanglement layers in the PQC allowed the algorithm to capture complex correlations between PWM signal features, such as frequency and amplitude variations. This was evident in the quantum kernel matric which demonstrated higher separability of signal classes compared to classical methods.

2. Observations on Training Efficiency

The hybrid quantum-classical workflow showed steady convergence during training. However, challenges included:

- Noise Sensitivity: On real quantum hardware, results were slightly degraded due to noise, with testing accuracy dropping by about 3%.
- Circuit Depth: Increasing the circuit depth improved feature extraction but lead to longer training time and greater susceptibility to errors.

3. Limitations and Areas for Improvement

- The algorithm's performance on noisy data suggests the need to enhance error mitigation techniques.

Scaling the algorithm to larger datasets and more complex PWM signals may require further optimization of circuit design and encoding strategies.

Chapter 6 Discussion

The QPEC algorithm represents a novel integration of quantum computing and classical machine learning techniques for PWM signal classification. This section discusses the implications of the results, highlights the algorithm's strengths and weaknesses, explores its broader significance, and identifies areas for future research.

6.1 Implication of the Results

The results demonstrate the QPEC algorithm's ability to classify PWM signals with high accuracy, outperforming traditional machine learning models such as Support Vector Machines (SVMs) and Random Forests. This highlights the effectiveness of quantum feature extraction, particularly the use of entanglement layers to capture complex correlations between signal attributes like frequency, amplitude, and duty cycle.

These findings suggest that hybrid quantum-classical approaches can address challenges in signal classification tasks where traditional methods struggle with subtle feature distinctions.

6.2 Strengths of the QPEC Algorithm

1. Enhanced feature Extraction: the parameterized quantum circuit (PQC) employed in QPEC successfully mapped PWM signal data into a higher-dimensional Hilbert space, enabling better separability of signal classes.
2. Generalization: The algorithm demonstrated strong generalization on unseen data, as evidenced by its balanced precision, recall, and F1-score on the testing dataset.
3. Scalability with Current Technology: By leveraging a hybrid approach, the QPEC algorithm mitigates the limitations of noisy intermediate-scale quantum (NISQ) devices. Classical optimizers effectively handle parameter tuning, reducing computational overhead on quantum hardware.

4. Demonstrated Practical Utility: The QPEC algorithm's ability to classify PWM signals accurately positions it as a viable tool for applications in device identification, fault detection, and signal processing.

6.3 Challenges and Limitations

Despite its success, the QPEC algorithm has certain limitations that warrant further investigation:

1. Noise Sensitivity: On real quantum hardware noise in the quantum circuits lead to about a 3% drop in testing accuracy. While noise mitigation techniques like zero-noise extrapolation were employed, their effectiveness was limited by the algorithm's circuit depth.
2. Circuit Depth and Resource Constraints: Increasing the circuit depth enhanced feature extraction but introduced trade-offs in training time and hardware requirements. This highlights the need for optimized circuit designs that balance expressivity with efficiency.
3. Encoding Complexity: The choice of amplitude and rotation encoding, while effective, may not scale well for larger datasets or more complex signals. Alternative encoding strategies should be explored to improve efficiency.
4. Limited Dataset Size: The PWM dataset used for this study, while representative, may not fully capture the diversity or real-world signal variations. Scaling the algorithm to larger, more diverse datasets is necessary to validate tis robustness.

6.4 Broader Significance

The QPEC algorithm introduces a groundbreaking approach to signal classification by leveraging quantum computing's unique properties. Its superior performance compared to classical models demonstrates the potential of hybrid quantum-classical methods to revolutionize fields such as:

- Device Authentication: Accurate identification of devices based on their PWM signals can enhance cybersecurity in IoT systems.
- Fault Detection: The ability to distinguish subtle signal anomalies makes QPEC valuable for predictive maintenance in critical systems.
- Signal Processing: Quantum-enhanced feature extraction may advance methodologies in communication and control systems.

Moreover, the study serves as a practical demonstration of how quantum computing can address real-world problems, bridging the gap between theoretical quantum algorithms and applied machine learning.

The proposed quantum algorithm for device identification based on PWM signal analysis has significant practical applications across various domains. By leveraging the computational advantages of quantum processing, this approach enhances the accuracy, efficiency, and robustness of device classification, particularly in scenarios of device classification, particularly in scenarios where classical methods may be computationally prohibitive or less effective in distinguishing fine-grained signal variations.

As the proliferation of Internet of Things (IoT) devices increases, ensuring secure authentication and preventing unauthorized access have become critical challenges. Traditional cryptographic approaches rely on key-based authentication, which is susceptible to key compromise and computational inefficiencies in resource-constrained environments. The proposed Quantum algorithm enables intrinsic device fingerprinting by extracting unique PWM signal distortions, allowing for non-invasive and hardware-based authentication mechanisms. This application enhances IoT security by mitigating spoofing and cloning attacks.

In industrial automation and aerospace systems, the ability to detect anomalies in embedded controllers is essential for predictive maintenance and fault prevention. By analyzing subtle variations in PWM signal characteristics, the quantum algorithm, can identify early-stage degradation in electronic components, actuators, or motor controllers. This capability facilitates proactive maintenance scheduling, reducing downtime and improving overall system reliability.

Modern smart home ecosystems consist of interconnected devices that communicate via embedded control systems. Efficient device classification and authentication are necessary to optimize energy consumption, network performance, and device interoperability. The quantum-enhanced classification of PWM signals enables precise device identification, improving system automation, reducing redundancy, and enhancing user experience in smart home applications.

Medical devices, such as infusion pumps, ventilators, and implantable sensors, often rely on PWM-driven actuators for precision control. Ensuring the integrity and authenticity of these devices is crucial for patient safety and regulatory compliance. The proposed quantum algorithm offers a non-invasive method for identifying medical devices based on the unique PWM signal characteristics, enabling continuous monitoring for potential malfunction, unauthorized device substitution, or security breaches.

The integration of quantum computing with machine learning techniques presents new opportunities in high-dimensional signal analysis. The quantum algorithm can improve feature extraction and classification efficiency for PWM signal data, particularly in real-time applications such as autonomous systems, robotics, and telecommunications. By leveraging quantum parallelism, the proposed approach has the potential to outperform classical machine learning methods in both speed and accuracy, especially for complex signal environments where traditional methods may struggle with scalability.

The practical applications outlined above demonstrate the potential impact of quantum-enhanced PWM signal analysis across multiple industries. By providing a novel, computationally efficient method for device identification and classification, this research contributes to advancements in security, automation, and reliability in embedded system applications.

6.5 Future Research Directions

Building on this work, several avenues for future research can be pursued:

1. Improved Error Mitigation: Investigate advanced error correction and noise reduction techniques to enhance the algorithm's robustness on quantum hardware.
2. Optimized Encoding Strategies: Explore alternative data encoding methods, such as tensor-product encodings or kernel-inspired encodings, to improve efficiency and scalability.
3. Adaptation to Larger Datasets: Test the QPEC algorithm, on larger and more diverse datasets to assess its scalability and performance in real-world scenarios.
4. Enhanced Circuit Design: Develop more efficient parameterized quantum circuits that reduce depth without sacrificing expressivity.
5. Integration with Emerging Hardware: As quantum hardware evolves, adapt the algorithm, to leverage advancements such as error-corrected qubits or hardware-specific optimizations.

Chapter 7 Conclusion

The QPEC algorithm showcases the potential of quantum computing to transform signal classification tasks by extracting features and modeling correlations that classical methods cannot easily capture. While challenges remain, the results of this study highlight a promising path forward for hybrid quantum-classical machine learning, opening doors to new applications and advancements in the field.

Bibliography

1. Nielsen, M. A., & Chuang, I. L. (2010). *Quantum Computation and Quantum Information*. Cambridge University Press.
2. Preskill, J. (2018). Quantum Computing in NISQ Era and Beyond. *Quantum,* 2, 79.
3. Schuld, M., & Petruccione, F. (2018). Supervised Learning with Quantum Computers. *Springer International Publishing.*
4. Gupta, R., Patel, S., & Singh, M. (2019). Applications of PWM Signal Distortion in Device Identification. *International Journal of Electrical and Computer Engineering*, 22(3), 201,210.
5. Montanero, A. (2016). Quantum Algorithms: An Overview. *npj Quantum Information,* **2(1),** 15023.
6. Wang, L., & Zhang, Q. (2018). The Influence of Parasitic Elements on PWM Signal Distortion. *Journal of Electrical Engineering and Technology,* 13(4), 1019-1025.
7. Havlicek, V., Corcoles, A. D., Temme, K., et al. (2019). Supervised Learning with Quantum-Enhanced Feature Spaces. *Nature*, 567(7747), 209-212.
8. Benedetti, M., Lloyd, E., Sack, S., & Fiorentini, M. (2019). Parameterized Quantum Circuits as Machine Learning Models. *Quantum Science and Technology*, 4(4), 043001.
9. Peruzzo, A., McClean, J., Shadbolt, P., et al. (2014). A Variational Eigenvalue Solver on a Photonic Quantum Processor. *Nature Communications*, 5(1), 4213.
10. Dunjko, V., & Briegel, H. J. (2018). Machine Learning & Artificial Intelligence in the Quantum Domain. Reports on Progress in Physics, 81(7), 074001.
11. Grant, E., Benedetti, M. Cao, S., et al. (2018). Hierarchical Quantum Classifiers*, npj Quantum Information*, 4(1), 65.

12. Lloyd, S., Mohseni, M., & Rebentrost, P. (2013). Quantum Algorithms for Supervised and unsupervised Machine Learning. *arXiv:1307.0411.*

13. Zhou, H., Chen P., &Li, J. (2021). Advanced Machine Learning Techniques for Signal Classification. *IEEE Access*, 9, 1563-1572.