CONTENT

CHAPTER -1 Quantum Computing: A Comprehensive Exploration.......01

1) Introduction
2) Quantum Computing Fundamentals
3) Quantum Gates and Circuits
4) Quantum Algorithm
5) Quantum Hardware
6) Quantum Error Correction
7) Application of Quantum Computing
8) Conclusion

CHAPTER – 2 Quantum Computing Technologies:

Foundations, Implementations, and Future Directions ……………………28

1) Introduction
2) Fundamental Concepts in Quantum Computing
3) Criteria for Evaluating Quantum Technologies
4) Overview of Quantum Computing Technologies
5) Comparative Analysis
6) Industry Landscape and Progress
7) Challenges and Limitations
8) Future Directions
9) Conclusion

CHAPTER - 3 Quantum Image Processing: A New Paradigm for Visual Computing ……………………………………………………………………35

1) Introduction
2) Research Status of Quantum Image Processing
3) Opportunities and Challenges
4) Conclusions

CHAPTER - 4 Quantum Image Classification: A Comprehensive Study………………………………………………………………………48

1) Introduction

2) Quantum Image Representation Models
3) Quantum Classification Algorithms
4) Experimental Evaluation
5) Comparative Analysis
6) Implementation Platforms
7) Challenges
8) Future Research Directions
9) Conclusion

CHAPTER – 5 Hybrid Quantum Neural Network……………………….53

1) Model
2) Application of Hybrid Quantum Models
3) Performance Comparison of HQNNs with CNNs
4) Challenges and Future Directions
5) Conclusion

Literature Review………………………………………………………..61

1) Quantum Computing Foundations
2) Image Processing and Classification Techniques
3) Quantum Approaches to Image Classification
4) Quantum Machine Learning (QML)
5) Hybrid Quantum Neural Networks (HQNN)
6) Challenges and Future Directions

References……………………………………………………………….65

CHAPTER -1 Quantum Computing: A Comprehensive Exploration

Abstract

Quantum computing, a transformative field that emerged from quantum mechanics and computer science, has gained immense attention for its potential to revolutionize computation. This paper aims to address the fundamentals of quantum computing and provide a comprehensive guide for both novices and experts in the field of quantum computing. Beginning with the foundational principles of quantum computing, we introduce readers to the fundamental concepts of qubits, superposition, entanglement, interference, and noise. We explore quantum hardware, quantum gates, and basic quantum circuits. This study offers insight into the current phase of quantum computing, including the noisy intermediate-scale quantum (NISQ) era and its potential for solving real-world problems. Furthermore, we discuss the development of quantum algorithms and their applications, with a focus on famous algorithms like Shor’s algorithm and Grover’s algorithm. We also touch upon quantum computing’s impact on various industries, such as cryptography, optimization, machine learning, and material science. By the end of this paper, readers will have a solid understanding of quantum computing’s principles, applications, and the steps involved in developing quantum circuits. Our goal is to provide a valuable resource for those eager to embark on their quantum computing journey and for researchers looking to stay updated on this rapidly evolving field.

1. Introduction

A **quantum computer** is a computer that exploits quantum mechanical phenomena. On small scales, physical matter exhibits properties of both particles and waves, and quantum computing takes advantage of this behaviour using specialized hardware. Classical physics cannot explain the operation of these quantum devices, and a scalable quantum computer could perform some calculations exponentially faster[a] than any modern "classical" computer. Theoretically a large-scale quantum computer could break some widely used encryption schemes and aid physicists in performing physical simulations; however, the current state of the art is largely experimental and impractical, with several obstacles to useful applications.

The basic unit of information in quantum computing, the qubit (or "quantum bit"), serves the same function as the bit in classical computing. However, unlike a classical bit, which can be in one of two states (a binary), a qubit can exist in a superposition of its two "basis" states, a state that is in an abstract sense "between" the two basis states. When measuring a qubit, the result is a probabilistic output of a classical bit. If a quantum computer manipulates the qubit in a particular way, wave interference effects can amplify the desired measurement results. The design of quantum algorithms involves creating procedures that allow a quantum computer to perform calculations efficiently and quickly.

Quantum computers are not yet practical for real-world applications. Physically engineering high-quality qubits has proven to be challenging. If a physical qubit is not sufficiently isolated from its environment, it suffers from quantum decoherence, introducing noise into calculations. National governments have invested heavily in experimental research aimed at developing scalable qubits with longer coherence times and lower error rates. Example implementations include superconductors (which isolate an electrical current by eliminating electrical resistance) and ion traps (which confine a single atomic particle using electromagnetic fields).

1.1 History

Quantum computing's origins trace back to the early 20th century with the development of quantum mechanics. Pioneers like Max Planck, Albert Einstein, and Niels Bohr made significant contributions. Erwin Schrödinger and Werner Heisenberg also played pivotal roles. They established the principles of superposition, entanglement, and wave-particle duality. These pioneers provided the foundation for understanding how quantum systems behave. While these ideas revolutionized physics, their computational implications remained unexplored for decades.

The link between quantum mechanics and computation was first recognized in the 1980s. In 1981, Richard Feynman proposed that classical computers struggle to efficiently simulate quantum systems. He suggested using a machine based on quantum mechanics. This machine would be better suited for this task (Feynman, International Journal of Theoretical Physics, 1982). Around the same time, Paul Benioff formulated a quantum mechanical model of a Turing machine. This model demonstrated that quantum mechanics could, in principle, be used for computation (Benioff, Journal of Statistical Physics, 1980).

1.1.1. Birth of Quantum Algorithms

In 1985, David Deutsch at the University of Oxford formulated the concept of a universal quantum computer. He defined quantum Turing machines. He introduced the idea of quantum parallelism. In this concept, a quantum system could process multiple possibilities simultaneously due to superposition (Deutsch, Proceedings of the Royal Society A, 1985). This theoretical framework set the stage for later quantum algorithms.

The first concrete demonstration of quantum computational advantage emerged in 1994 when Peter Shor developed a quantum algorithm capable of factoring large numbers exponentially faster than any known classical algorithm. Shor's algorithm threatened classical cryptographic systems, particularly RSA encryption, which relies on the difficulty of prime factorization (Shor, Proceedings of the 35th Annual Symposium on Foundations of Computer Science, 1994). Around the same time, Lov Grover introduced an algorithm in 1996 that provided a quadratic speedup for searching unsorted databases (Grover, Proceedings of the 28th Annual ACM Symposium on Theory of Computing, 1996).

1.2.2 Early Beginnings Of Quantum Mechanics

The early beginnings of quantum mechanics can be traced back to the late 19th century when scientists such as Max Planck and Albert Einstein began questioning the fundamental principles of classical physics. In 1900, Planck introduced the concept of the "quantum" in his work on black-body radiation, proposing that energy is not continuous but comes in discrete packets, or quanta (Planck, 1901). This idea was revolutionary then and laid the foundation for developing quantum theory.

In the early 20th century, scientists such as Niels Bohr and Louis de Broglie built upon Planck's work, introducing new concepts such as wave-particle duality and the uncertainty principle. In 1913, Bohr proposed his atom model, positing that electrons occupy specific energy levels, or shells, around the nucleus (Bohr, 1913). This model significantly improved over earlier atomic models and helped explain many experimental observations.

1.2.3. Development Of Quantum Theory Foundations

The development of quantum theory foundations began in the early 20th century, with Max Planck's introduction of the concept of quantized energy in 1900 (Planck, 1901). This idea challenged the traditional understanding of energy as a continuous variable and laid the groundwork for developing quantum mechanics. In 1905, Albert Einstein further developed this concept by introducing the idea of light quanta, now known as photons, which have both wave-like and particle-like properties (Einstein, 1905).

The next major milestone in the development of quantum theory foundations was the introduction of the Bohr model of the atom in 1913 (Bohr, 1913). Niels Bohr's model posited that electrons occupy specific energy levels, or shells, around the nucleus and can jump from one level to another by emitting or absorbing energy. This model significantly improved over earlier atomic models but still had limitations.

1.2.4. Introduction To Quantum Information Science

Quantum information science is an interdisciplinary field that combines principles from physics, mathematics, computer science, and engineering to study the behavior of quantum systems and their potential applications in information processing. The concept of qubits, or quantum bits, was first introduced by physicist Paul Benioff in 1980 as a theoretical framework for quantum computing (Benioff, 1980). Qubits are unique because they can exist in multiple states simultaneously, allowing for the simultaneous processing of vast amounts of information.

The no-cloning theorem, proved by physicists Wootters and Zurek in 1982, is a fundamental concept in quantum information science. It states it is impossible to create a perfect copy of an arbitrary qubit (Wootters & Zurek, 1982). This theorem has significant implications for developing quantum cryptography and quantum computing. Quantum entanglement, another key concept in quantum information science, was first described by Albert Einstein, Boris Podolsky, and Nathan Rosen in their famous EPR paper in 1935 (Einstein et al., 1935).

1.2.5. First Quantum Computer Proposals Emerged

The concept of quantum computing dates back to the early 1980s, when physicist Paul Benioff proposed the idea of a quantum mechanical model of computation. This proposal was followed by David Deutsch's 1985 paper, "Quantum Theory, the Church-Turing Principle and the universal quantum computer," which introduced the concept of a universal quantum computer.

In his paper, Deutsch described a theoretical model for a quantum computer that could solve problems exponentially faster than classical computers. He also proposed the idea of quantum parallelism, where a single quantum computer could perform many calculations simultaneously. Other researchers, including Richard Feynman and Yuri Manin, later developed this idea.

1.2.6. David Deutsch's Quantum Turing Machine

The concept of the Quantum Turing Machine (QTM) was first introduced by David Deutsch in his 1985 paper "Quantum theory, the Church-Turing Principle and the universal quantum computer". In this work, Deutsch proposed a theoretical model for a quantum computer that could simulate any physical system, including itself. The QTM is based on the idea of a Turing machine, which is a mathematical model for computation developed by Alan Turing in the 1930s.

The QTM consists of a read/write head that moves along an infinite tape divided into cells, each of which can hold a qubit (quantum bit) of information. The QTM's operation is governed by a set of quantum gates, which are the quantum equivalent of logic gates in classical computing. These gates perform superposition, entanglement, and measurement on the qubits stored in the cells. Deutsch showed that a QTM could simulate any physical system with sufficient resources.

1.2.7. Shor's Algorithm For Factorization Discovered

Shor's algorithm for factorization was discovered in 1994 by mathematician Peter Shor, who was working at Bell Labs then. The algorithm is a quantum algorithm that can factor large numbers exponentially faster than any known classical algorithm. This discovery was significant because it showed that quantum computers could solve specific problems much more efficiently than classical computers.

The algorithm uses quantum parallelism and interference. It finds the period of a function related to the number being factored. This period is then used to factor the number using the Pollard's rho algorithm. The key insight behind Shor's algorithm was to use the Quantum Fourier transform (QFT) to efficiently compute the discrete Fourier transform of the function, allowing for the period extraction.

1.2.8. Quantum Error Correction Codes Developed

Quantum Error Correction Codes (QECCs) are crucial for developing reliable quantum computers. One of the earliest QECCs is the Shor code, proposed by Peter Shor in 1995. This code encodes a single qubit into nine physical qubits and can correct any single-qubit error. The Shor code uses a combination of bit-flip and phase-flip errors to detect and correct errors.

Another important QECC is the Steane code, developed by Andrew Steane in 1996. This code encodes a single qubit into seven physical qubits and can also correct any single-qubit error. The Steane code uses a combination of bit-flip and phase-flip errors to detect and correct errors, similar to the Shor code.

1.2.9. First Experimental Quantum Computers Built

The first experimental quantum computers were built in the late 1990s, with IBM's 3-qubit quantum computer being one of the earliest examples. This computer used nuclear magnetic resonance (NMR) to manipulate the qubits, which are the fundamental units of quantum information. The NMR technique allowed researchers to control the spin of atomic nuclei, effectively creating a quantum bit that could exist in multiple states simultaneously.

One of the key challenges in building these early quantum computers was maintaining control over the fragile quantum states. Researchers had to develop new techniques for error correction and noise reduction, as even slight disturbances could cause the qubits to lose their quantum properties. For example, a study published in the journal Nature in 1998 demonstrated the use of a technique called "quantum error correction" to protect against decoherence, which is the loss of quantum coherence due to environmental interactions.

1.2.10. Quantum Computing Breakthroughs In 2000s

In the early 2000s, significant breakthroughs were made in quantum computing, particularly in the development of quantum algorithms and the demonstration of quantum supremacy. One notable achievement was the implementation of Shor's algorithm for factorizing large numbers on a small-scale quantum computer (Vandersypen et al., 2001). This algorithm, proposed by Peter Shor in 1994, demonstrated the potential power of quantum computing over classical computing for certain problems.

Another important breakthrough came with the development of the Quantum Approximate Optimization Algorithm (QAOA) by Edward Farhi and collaborators in 2014. However, its roots date back to the early 2000s when similar ideas were explored (Farhi et al., 2000). QAOA is a hybrid quantum-classical algorithm designed for solving optimization problems more efficiently than classical algorithms.

1.2.11. Development Of Topological Quantum Computing

Topological quantum computing is an approach to building a quantum computer that uses exotic states of matter called topological phases to store and manipulate quantum information. This approach was first proposed by physicist Alexei Kitaev in 1997, who showed that certain types of topological phases could be used to create robust and fault-tolerant quantum computers (Kitaev, 2003). The idea is to use the non-Abelian anyons that arise in these systems as a basis for quantum computation. Non-Abelian anyons are exotic quasiparticles that can be used to store and manipulate quantum information in a way that is inherently fault-tolerant.

One of the key advantages of topological quantum computing is its potential for robustness against decoherence, which is the loss of quantum coherence due to interactions with the environment. Topological phases are characterized by their ability to exhibit non-Abelian statistics, which means that the exchange of two anyons can result in a non-trivial transformation of the system's state (Nayak et al., 2008). This property makes it possible to use topological phases as a basis for quantum computation that is inherently robust against decoherence.

1.2.12. Recent Advances In Quantum Computing Hardware

In May 2023, Quantinuum introduced the System Model H2, achieving a quantum volume of 65,536, the largest on record. This system demonstrated significant advancements in qubit stability and error correction, essential for reliable quantum computations.

Recent developments in quantum computing have typically focused on increasing the number of qubits and improving their stability. In December 2023, IBM unveiled the IBM Condor, a 1,121-qubit processor, marking a significant milestone in scaling quantum processors. This processor features a 50% increase in qubit density compared to its predecessor, the IBM Osprey.

In November 2024, Microsoft and Atom Computing announced a significant milestone in quantum computing by creating and entangling 24 logical qubits using neutral atoms. This achievement also included the demonstration of error detection and correction capabilities on 28 logical qubits, marking a substantial step toward reliable quantum computation.

In December 2024, Google unveiled the "Willow" processor, a 105-qubit superconducting quantum chip that achieved below-threshold quantum error correction. This processor demonstrated an exponential reduction in errors as the number of qubits increased, marking a significant milestone in enhancing qubit stability.

2. Quantum Computing Fundamentals

In lieu of bits in classical computing, quantum computing utilizes qubits, which can exist in multiple states at the same time—a phenomenon known as *superposition*. Quantum *entanglement* signifies a unique connection between qubits and quantum *interference* can alter the outcome of the qubits. Quantum computers also face a challenge called quantum *noise*, which can lead to loss of quantum properties, such as superposition, entanglement, and interference, and can affect the outcome of a quantum system. This section aims to provide a simple yet comprehensive understanding of quantum computing fundamentals.

2.1. Qubits

A qubit, or quantum bit, is the basic unit of information used to encode data in quantum computing and can be best understood as the quantum equivalent of the traditional bit used by classical computers to encode information in binary.

The term “qubit” is attributed to American theoretical physicist Benjamin Schumacher. Qubits are generally, although not exclusively, created by manipulating and measuring quantum particles (the smallest known building blocks of the physical universe), such as photons, electrons, trapped ions, superconducting circuits, and atoms.

Enabled by the unique properties of quantum mechanics, quantum computers use qubits to store more data than traditional bits, vastly improve cryptographic systems, and perform very advanced computations that would take thousands of years (or be impossible) for even classical supercomputers to complete.

Powered by qubits, quantum computers may soon prove pivotal in addressing many of humanity’s greatest challenges, including cancer and other medical research, climate change, machine learning and artificial intelligence (AI).

2.1.1. Bra-Ket Notation

Qubit is a quantum computing particle that has a wave-like nature with wavefunction $\psi(x)$ that satisfies the Schro¨ dinger equation. Theoretically, this wavefunction exists in an infinite dimensional Hilbert dual space. Therefore, the state vector representing this wavefunction in Hilbert space requires an infinite dimensional vector notation. This infinite dimensional vector state of the qubit in Hilbert dual space is shown using Dirac’s bra-ket notation, which was created by Paul Dirac in 1939 . However, it can also be a finite-dimensional vector having two states, on/off or spin-up/spin-down, which can be shown in two-dimensional Hilbert space. In this notation, two-dimensional state vectors $|1\rangle$ (read ket one) and $|0\rangle$ (read ket zero) are used for qubit.

$|0\rangle|1\rangle=1|0\rangle+0|1\rangle\rightarrow[10]=0|0\rangle+1|1\rangle\rightarrow[01](1)(2)$

In the equation (1), ket zero shows that the qubit is at an off or spin-down state. Here, the first element represents the probability amplitude of off or spin down, and the second element shows the probability amplitude of on or spin up. Probability amplitude can be a complex value and it is used to compute the probabilities of vector states. Additionally, in Dirac’s notation, the bra is a complex conjugate transpose of a ket. For example, $\langle\phi|$ (read bra of ϕ) is a complex conjugate transpose vector of ket ψ . The inner product of these two vectors $\langle\phi|\psi\rangle$ is a scalar value . The symbol “$|\rangle$ ” denotes a column vector, and is known as a “ket”. The “bra” ($\langle|$) form is a row vector and it is shown below:

⟨0|⟨1|=1⟨0|+0⟨1|→[10]=0⟨0|+1⟨1|→[01](3)(4)

The ket notation is widely used in quantum computing as bra-ket representation of the qubit. The following two states (|0⟩+|1⟩)/2–√ and (|0⟩−|1⟩)/2–√ are also commonly used in quantum calculations and these are sometimes written as |+⟩ and |−⟩ , respectively.

A single qubit is also called a two-level quantum system because it is a linear combination of two state basis, 0 and 1. Below is the common form of a single qubit in bra-ket notation.

|v⟩=v0|0⟩+v1|1⟩=v0[10]+v1[01]=[v0v1](5)

Here vo and v1 are complex coefficients to measure probability amplitudes. The probability and the phase of each computational state basis for a qubit can be computed as follows:

For state basis |0⟩ with complex coefficient vo=x+i∗y

probabilityamplitude|vo|probabilityphase(rad)phase(degree)=|vo|=(x+i⋅y)∗(x−i⋅y)−−−−−−−−−−−−−−−−−√=x2+y2−−−−−−√=|vo|2=tan−1yx=phase(rad)∗(180/π)(6)(7)(8)(9)(10)

The probability amplitude is used to calculate the probability of each state basis of the qubit which helps in the measurement of the qubit state. Similarly, phase is used for quantifying *interference*. The concepts of measurement and interference are explained in the following sections of the paper. If the complex coefficients are normalized, then they represent the probability of the qubit for 0 and 1 state

|vo|2+|v1|2=1(11)

This is known as the **normalization constraint** since all two-level systems must obey this quality to function as a qubit.

For two or multiple qubits, the tensor product (or Kronecker product) is used to compute the resultant states of the quantum system. The tensor product is denoted by the symbol ⊗ . Let us consider two qubits |a⟩ and |b⟩ as

|a⟩=[a0a1]and|b⟩=[b0b1](12)

The tensor product of the two qubits is

|x⟩|x⟩|x⟩|x⟩=|a⟩⊗|b⟩=|ab⟩=⎡⎣⎢⎢⎢⎢⎢a0∗[b0b1]a1∗[b0b1]⎤⎦⎥⎥⎥⎥⎥=⎡⎣⎢⎢⎢a0b0a0b1a1b0a1b1⎤⎦⎥⎥⎥=⎡⎣⎢⎢⎢x0x1x2x3⎤⎦⎥⎥⎥=a0b0|00⟩+a0b1|01⟩+a1b0|10⟩+a1b1|11⟩=x0|00⟩+x1|01⟩+x2|10⟩+x3|11⟩(13)(14)(15)(16)

and the **normalization constraint rule** for the two qubits will be the same as follows:

$$|a_0b_0|^2+|a_0b_1|^2+|a_1b_0|^2+|a_1b_1|^2=1 \quad (17)$$

Similarly for 3-qubits, if $|c\rangle=[c_0c_1]$ then the tensor product of the three qubits is

$$|\gamma\rangle|\gamma\rangle=|ab\rangle\otimes|c\rangle=|abc\rangle=\begin{bmatrix} a_0b_0*[c_0c_1] \\ a_0b_1*[c_0c_1] \\ a_1b_0*[c_0c_1] \\ a_1b_1*[c_0c_1] \end{bmatrix}=\begin{bmatrix} a_0b_oc_0 \\ a_0b_oc_1 \\ a_0b_1c_0 \\ a_0b_1c_1 \\ a_1b_oc_0 \\ a_1b_oc_1 \\ a_1b_1c_0 \\ a_1b_1c_1 \end{bmatrix}=\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} \quad (18)(19)$$

The same method will be used to combine n qubits, and normalization constraint rules for n -qubits will be given as in.

$$\sum|v_i|^2=1 \quad (20)$$

If we have n qubits, we will need to keep track of 2n complex probability amplitudes. As we can see, these vectors grow exponentially with the number of qubits. This is the reason quantum computers with large numbers of qubits are so difficult to simulate in classical computers. A modern laptop can easily simulate a general quantum state of around 20 qubits, but simulating 100 qubits is too difficult even for the largest supercomputers.

2.1.2. Bloch Sphere Notation

The Bloch sphere is a mathematical representation of a given quantum state of a qubit, with which researchers can pinpoint and manipulate various such states within the sphere to their advantage. Three qubits $|1\rangle$ $|-\rangle$ and, $|\gamma\rangle$ are shown in Bloch sphere representation in Figure 1.

Bloch sphere representation of three different qubits.

2.2. Quantum Superposition

Quantum superposition is a fundamental concept in quantum mechanics, the branch of physics that deals with the behavior of particles at the smallest scales. At its core, superposition means that a quantum system, such as an electron, can exist in multiple states simultaneously.

Consider an electron, the tiny negatively charged particle that orbits the nucleus of an atom. In classical physics, we might expect it to have a well-defined position and velocity at any given moment. However, in the quantum world, things get weird. An electron can be in a superposition of multiple positions and velocities simultaneously.

Mathematically, we represent superposition using a wave function, denoted as Ψ (psi). The wave function describes the probability distribution of finding the electron in various states. When we measure the electron's position, it "collapses" into one of those states, and we observe a definite position.

Here's the fascinating part: until we make that measurement, the electron exists in all possible states simultaneously. It's as if it's exploring all its options at once, like Schrödinger's cat being both alive and dead until observed (which is another intriguing quantum concept).

2.3. Quantum Entanglement

Quantum entanglement is one of the foundational properties of quantum mechanics and quantum technology. It is a phenomenon that explains how two subatomic particles can be intimately linked to each other so that two particles behave like a single unified unit, irrespective of their distance. The Nobel Prize in 2022 for Physics was given for the experimental establishment of the reality of quantum entanglement.

Quantum entanglement properties can be exploited to open up new technological possibilities such as quantum cryptography, superdense coding, and teleportation to solve modern-day problems.

2.3.1. Discovery of Quantum Entanglement Phenomenon

In the early 20th century, Physicists developed the basic concepts behind entanglement once they figured out the mechanics of the quantum world.

Einstein–Podolsky–Rosen paradox: Albert Einstein, Boris Podolsky and Nathan Rosen (1935), examined the correlationof quantum states that would interact with each other, a phenomenon that came to be later known as "Entanglement".

They discovered that the strongly correlated particles lose their individual quantum states and instead share a single, unified state.

Schrödinger's paper: Erwin Schrödinger was the first one to use the word "entanglement".

According to him, entanglement is the most significant aspect of quantum mechanics that allows for the transmission of information at speeds faster than light.

Albert Einstein termed this entanglement as ‘spooky action at a distance’.

Nobel Prize in Physics 2022: Alain Aspect, John Clauser and Anton Zeilinger were conferred the prize for the experimental establishment of the reality of quantum entanglement.

The demonstration of the ability to control the teleportation of entangled particles paved the way for the foundation of 21st-century emerging quantum technology.

2.3.2. How does Quantum Entanglement work?

Quantum entanglement is a phenomenon in quantum physics where two or more particles become interconnected in such a way that the state of one particle instantly influences the state of the other, regardless of the distance between them.

For example, consider a pair of entangled photons with random polarisations.

When the polarisation of one photon is measured and found to be vertical, the entanglement ensures that the polarization of the other photon will also be vertical.

When an operation is performed on one of the entangled particles, there is an instantaneous reaction on the other. Hence, measurements of one state can affect the other.

2.3.3. Methods to Create Entangled Particles

There are many ways of creating pairs of entangled particles, such as photons.

Special crystals method: A photon with high energy is converted into two photons of lower energy, this is known as “down conversion”. This allows large numbers of entangled photon pairs to be produced quickly and easily.

Cooling of the particles: One method is to cool the particles and place them close enough together so that their quantum states overlap, making it possible for identical particles.

Nuclear decay: Using the subatomic under the process of nuclear decay, entangled particles can be produced automatically.

Splitting an individual photon: The entangled pairs of photons can also be produced by splitting a single photon into two, or by mixing two pairs of photons in an optical fibre cable.

2.3.4. Device-Independent Self-Testing (DIST) Method

Due to the fragile nature of entanglement, characteristics of particles during the transit can be lost. Hence, it is extremely important to verify.

The DIST methodis a novel protocol developed by scientists from S. N. Bose National Centre for Basic Sciences (SNBNCBS), Kolkata.

This method enables the measurement of the status of entanglement in an unknown quantum state of two particles.

It is a device-independent method and overcomes the possibility of being hacked or compromised.

2.3.5. Applications of Quantum Entanglement

The predictability of entangled particles (achieved by laws of Quantum) is employed to make technological progress, construct quantum computers, improve measurements, build quantum networks, establish secure quantum encrypted communication etc.

Quantum Cryptography: The entangled particles are used for secure communication.

It prevents the leakages of information by changing particle states if someone tries to intercept them.

The entanglement-based quantum distributed keys are being used to improve the security of communications.

Improved Microscopy: The entangled photons help in making microscopes sharper and gathering more detailed information.

It boosts medical imaging through optical sensing.

This helps in various scientific fields, like biophysical characterisation of cells as well as in nanotechnology.

Quantum Teleportation: Quantum teleportation involves the exchange of quantum information between entangled particles.

It helps quantum computers work faster and consume less power.

Superdense Coding: The basic reason behind superdense coding is to send more information using fewer particles.

It makes data transmission more efficient and faster.

Application in defence: Quantum communications, quantum computing and quantum sensing technologies which are based on entanglement can be applied by military and intelligence agencies.

Improving GPS positioning: It can improve the high resolution radio frequency detection thereby improving the GPS positioning.

Entanglement-as-a-service: The importance of quantum entanglement is such that nowadays, there is hot debate about entanglement-as-a-service (EAAS) is going on.

By this, a company would provide customers with network access to entangled qubits to be used for secure communications.

2.3.6. Challenges Associated with Quantum Entanglement

Quantum entanglement as a technological resource can not be used directly even though the actual experimental techniques pioneered the entanglement and other properties. It only developed as a foundation for using quantum entanglement as a technological resource.

Fragility: Entanglement is fragile and is easily lost during the transit of photons through the environment. Hence using them as a resource is extremely challenging.

Observance: Quantum superposition and entanglement only exist as long as quantum particles are not observed or measured. Observing particles leads to a collapse of the system.

Practical problems: Researchers have revealed (on simulating a wormhole model)

some practical limitations like finding the shape of an undiscovered drug, autonomously exploring space or factoring large numbers.

Vulnerable to physical shock: Researchers are yet to build QCs that completely eliminate these disturbances in systems as Qubits exist in superposition in specific conditions, such as low temperature (~0.01 K), with radiation shielding and protection against physical shock....

2.4. Quantum Interference

Qubit is represented with bra-ket notation or Bloch sphere but this is just a mathematical representation of the qubit state. In reality, the qubit has a wave-like nature that is described by a quantum wavefunction satisfying the Schro¨ dinger equation as shown in Figure 4. A wavefunction is a mathematical description of the quantum state that consists of complex probability amplitudes, and the corresponding probabilities of quantum system states.

Wave-like nature of a qubit.

When we have multiple qubits, their wavefunctions are added together to give an overall wavefunction describing the resultant states of a quantum system. This adding process of wavefunctions is called interference. It is a fundamental phenomenon that arises from the wave-like nature of quantum particles, such as electrons or photons and it distinguishes quantum systems from classical systems.

In quantum computing, when two quantum wavefunctions overlap, they can interfere with each other constructively or destructively. This results in a change in the resultant wavefunction of the quantum system that affects the probability distribution of its quantum states as shown in Figure 5.

Quantum interference in two quantum wavefunctions.

Interference can also be a challenge in quantum computing due to the phenomenon of decoherence. Decoherence is the loss of quantum coherence, which is the property of a quantum state to maintain its superposition and entanglement, due to the interactions with environment and thermalization. This loss of coherence leads to a breakdown of interference effects and making quantum computation error-prone. Quantum error correction techniques are used to mitigate the impact of the external environment and preserve the delicate quantum interference necessary for quantum computation.

Overall, interference is a foundational concept in quantum computing, allowing quantum systems to perform computations by updating the probability distributions of the quantum states. This concept solves certain problems in ways that are not achievable using classical computing methods.

2.5. Quantum Noise

Quantum noise refers to the uncertainty and fluctuations that arise in quantum systems due to the probabilistic nature of quantum mechanics. It is a challenge in quantum systems even at low temperatures.

In classical systems, noise is often associated with random variations in signals or disturbances caused by external factors. When a quantum system is in a superposition state, its outcome upon measurement is not deterministic but is determined by the probability distribution of the quantum states. Noise and error can affect the outcome due to the quantum system's interaction with the external environment. It can lead to loss of quantum properties (superposition, entanglement, and, interference) over time affecting the outcome of quantum circuits.

Quantum noise has several manifestations in quantum systems, and it can impact various aspects of quantum computing. Some common examples of quantum noise include:

Measurement Noise: When measuring a quantum system, the act of measurement can cause a quantum system to lose the quantum superposition and collapse the quantum state into one of its states, introducing uncertainty in the outcome due to the probabilistic nature of the measurement process.

Decoherence: Interactions with the environment can cause quantum systems to lose quantum superposition, entanglement, and interference, affecting the performance of quantum algorithms.

Quantum noise poses a significant challenge for quantum computing. To address this challenge, researchers have been working on quantum error correction techniques, which are essential for preserving the quantum states against the detrimental effects of measurement noise and decoherence.

3. Quantum Gates and Circuits

A **quantum (logic) gate** is a device which performs a fixed unitary operation on selected qubits in a fixed period of time, and a **quantum circuit** is a device consisting of quantum logic gates whose computational steps are synchronised in time.

The **size** of such a circuit is the number of gates it contains. The gates in a circuit can be divided into layers, where the gates in the same layer operate at the same time, and the number of such layers is called the **depth** of a circuit.

3.1 Quantum Logic Gates

In quantum computing and specifically the quantum circuit model of computation, a **quantum logic gate** (or simply **quantum gate**) is a basic quantum circuit operating on a small number of qubits. Quantum logic gates are the building blocks of quantum circuits, like classical logic gates are for conventional digital circuits.

Unlike many classical logic gates, quantum logic gates are reversible. It is possible to perform classical computing using only reversible gates. For example, the reversible Toffoli gate can implement all Boolean functions, often at the cost of having to use ancilla bits. The Toffoli gate has a direct quantum equivalent, showing that quantum circuits can perform all operations performed by classical circuits.

Quantum gates are unitary operators, and are described as unitary matrices relative to some orthonormal basis. Usually the *computational basis* is used, which unless comparing it with something, just means that for a *d*-level quantum system (such as a qubit, a quantum register, or qutrits and qudits)[1]: 22–23 the orthonormal basis vectors are labeled , or use binary notation.

Quantum gates manipulate qubits through unitary operations. Key gates include:

Hadamard (H): Creates superposition.

Pauli-X (NOT): Bit-flips the qubit.

CNOT: Entangles two qubits.

T and S Gates: Phase shift gates.

3.3 Quantum Circuits

In quantum information theory, a **quantum circuit** is a model for quantum computation, similar to classical circuits, in which a computation is a sequence of quantum gates, measurements, initializations of qubits to known values, and possibly other actions. The minimum set of actions that a circuit needs to be able to perform on the qubits to enable quantum computation is known as DiVincenzo's criteria.

Circuits are written such that the horizontal axis is time, starting at the left hand side and ending at the right. Horizontal lines are qubits, doubled lines represent classical bits. The items that are connected by these lines are operations performed on the qubits, such as measurements or gates. These lines define the sequence of events, and are usually not physical cables.

The graphical depiction of quantum circuit elements is described using a variant of the Penrose graphical notation. Richard Feynman used an early version of the quantum circuit notation in 1986.

Components:

Qubits: Represented as wires.

Gates: Placed on the wires to manipulate the state.

Measurements: At the end of the circuit to extract classical information.

Example Circuit:

A simple circuit that creates entanglement:

|0> ——H——■—— (Apply Hadamard then CNOT)

|

|0> ————X—— (Target qubit)

4. Quantum algorithm

In quantum computing, a **quantum algorithm** is an algorithm that runs on a realistic model of quantum computation, the most commonly used model being the quantum circuit model of computation. A classical (or non-quantum) algorithm is a finite sequence of instructions, or a step-by-step procedure for solving a problem, where each step or instruction can be performed on a classical computer. Similarly, a quantum algorithm is a step-by-step procedure, where each of the steps can be performed on a quantum computer. Although all classical algorithms can also be performed on a quantum computer, the term quantum algorithm is generally reserved for algorithms that seem inherently quantum, or use some essential feature of quantum computation such as quantum superposition or quantum entanglement.

Problems that are undecidable using classical computers remain undecidable using quantum computers. What makes quantum algorithms interesting is that they might be able to solve some problems faster than classical algorithms because the quantum superposition and quantum entanglement that quantum algorithms exploit generally cannot be efficiently simulated on classical computers (see Quantum supremacy).

The best-known algorithms are Shor's algorithm for factoring and Grover's algorithm for searching an unstructured database or an unordered list. Shor's algorithm runs much (almost exponentially) faster than the most efficient known classical algorithm for factoring, the general number field sieve. Grover's algorithm runs quadratically faster than the best possible classical algorithm for the same task, a linear search.

4.1. General Structure of a Quantum Algorithm

1. **Initialize** qubits (often to |0⟩).
2. **Apply quantum gates** (like H, X, CNOT) to create and manipulate quantum states.
3. **Use quantum subroutines**, like the Quantum Fourier Transform or Grover's Oracle.
4. **Measure** the qubits to get the classical output.

4.2. Examples of Quantum Algorithms

Shor's algorithm

Shor's algorithm solves the discrete logarithm problem and the integer factorization problem in polynomial time,[9] whereas the best known classical algorithms take super-polynomial time. It is unknown whether these problems are in P or NP-complete. It is also one of the few quantum algorithms that solves a non-black-box problem in polynomial time, where the best known classical algorithms run in super-polynomial time.

Grover's algorithm

Grover's algorithm searches an unstructured database (or an unordered list) with N entries for a marked entry, using only queries instead of the queries required classically. Classically, queries are required even allowing bounded-error probabilistic algorithms.

Theorists have considered a hypothetical generalization of a standard quantum computer that could access the histories of the hidden variables in Bohmian mechanics. (Such a computer is completely hypothetical and would *not* be a standard quantum computer, or even possible under the standard theory of quantum mechanics.) Such a hypothetical computer could implement a search of an N-item database in at most steps. This is slightly faster than the steps taken by Grover's algorithm. However, neither search method would allow either model of quantum computer to solve NP-complete problems in polynomial time.

Quantum Fourier Transform (QFT)

In quantum computing, the **quantum Fourier transform (QFT)** is a linear transformation on quantum bits, and is the quantum analogue of the discrete Fourier transform. The quantum Fourier transform is a part of many quantum algorithms, notably Shor's algorithm for factoring and computing the discrete logarithm, the quantum phase estimation algorithm for estimating the eigenvalues of a unitary operator, and algorithms for the hidden subgroup problem. The quantum Fourier transform was discovered by Don Coppersmith.[1] With small modifications to the QFT, it can also be used for performing fast integer arithmetic operations such as addition and multiplication.

The quantum Fourier transform can be performed efficiently on a quantum computer with a decomposition into the product of simpler unitary matrices. The discrete Fourier transform on amplitudes can be implemented as a quantum circuit consisting of only Hadamard gates and controlled phase shift gates, where is the number of qubits. This can be compared with the classical discrete Fourier transform, which takes gates (where is the number of bits), which is exponentially more than .

Variational Quantum Algorithms

The emergence of quantum computation as a solution to high computational complexity problems on classical devices, such as large-scale linear algebra and discrete optimization, has led to a key technological question: How can today's NISQ (Noisy Intermediate Scale Quantum) devices be optimally utilized to achieve quantum advantage?

In addressing this question, Variational Quantum Algorithms have emerged as promising solutions that respect the aforementioned constraints. These algorithms adopt the concept of training quantum computers in a similar manner to training neural networks, that is, finding the optimum parameters of the model to minimize/maximize some objective function related to that model.

5.Quantum Hardware

Quantum hardware refers to the physical components and systems used to build quantum computers—machines that harness the laws of quantum mechanics to perform computations. Unlike classical computers, which use bits (0 or 1), quantum computers use qubits, which can exist in superposition and be entangled. Building and maintaining these fragile states require specialized, high-precision hardware across multiple domains: physics, cryogenics, materials science, electronics, and computer engineering.

5.1 Types of Quantum Hardware

Quantum Hardware Types in Quantum Computing is a significant chapter because it delves into the physical foundations that enable quantum computers to operate. This chapter explores the various types of hardware architectures developed to harness quantum mechanics for computational purposes. While traditional computers rely on classical bits, quantum computers use quantum bits (qubits), which are incredibly sensitive and require specialized hardware to manage their unique properties like superposition and entanglement. Each type of quantum hardware has different advantages, limitations, and engineering challenges.

Superconducting Qubits

Description: Superconducting qubits are one of the most developed and widely used quantum hardware types. They are based on superconducting materials that can carry current without resistance when cooled to near absolute zero temperatures. This technology uses Josephson junctions—specialized circuits where quantum effects can create, manipulate, and read qubits.

How It Works:

Superconducting qubits are formed using circuits that can generate and control quantum states by applying microwave pulses.

These pulses cause the qubit to enter superposition, where it can represent both 0 and 1 simultaneously.

Quantum gates are implemented by manipulating these states through precise pulses and measurements.

Trapped Ion Qubits

Description: Trapped ion quantum computers use ions (charged atoms) as qubits, held in place by electromagnetic fields in a vacuum. The internal energy levels of the ions represent qubit states, and laser pulses are used to manipulate these states.

How It Works:

Ions are isolated in electromagnetic traps, where they remain stable.

Qubit states (0 and 1) are represented by different internal energy levels of the ion.

Quantum operations are performed by laser pulses that alter these states, with the ions interacting via the collective motion in the trap.

Topological Qubits

Description: Topological qubits are a promising but experimental form of quantum computing. They leverage anyons, exotic particles that exist in two-dimensional space, which can maintain quantum states in a way that is naturally resistant to errors. This error-resistance is a property of the “topology” of the system rather than of individual particle states.

How It Works:

Topological qubits store information in the braiding of anyons, where the paths of particles relative to each other create quantum states.

By entangling particles through specific paths, qubits are less sensitive to local disturbances, reducing errors.

Photonic Qubits

Description: Photonic quantum computers use particles of light (photons) as qubits. Since photons are less susceptible to environmental interference, they offer a different approach to building stable quantum computers that can operate at room temperature.

How It Works:

Photonic qubits are encoded in properties of photons, such as polarization or path.

Quantum gates and circuits are designed using optical components like beam splitters and phase shifters, which control and measure photons.

Neutral Atom Qubits

Description: Neutral atom quantum computers use individual atoms held in place by laser traps (optical tweezers) as qubits. These atoms are manipulated using laser pulses to perform quantum operations.

How It Works:

Atoms are trapped in an array and can be individually addressed with lasers.

Quantum gates are executed by inducing interactions between nearby atoms using Rydberg states, where an electron is excited to a high energy level.

Quantum Dots

Description: Quantum dots are nanoscale semiconductor particles that confine electrons or holes, acting as artificial atoms. Their quantized energy levels can represent qubit states.

How It Works:

Quantum dots trap individual electrons, and the electron’s spin state represents the qubit.

Manipulating the electron's spin using electric or magnetic fields allows control of the quantum states.

NV Centers in Diamond

Description: Nitrogen-vacancy (NV) centers in diamond are a solid-state approach where qubits are formed by imperfections in the diamond's crystal structure. The NV center, a nitrogen atom next to a vacant site in diamond, can hold quantum information.

How It Works:

The spin states of electrons in the NV center represent qubits.

By applying magnetic fields and microwave radiation, the spin states are manipulated to perform quantum operations.

Quantum Annealers

Description: Quantum annealers are a specialized type of quantum hardware designed for optimization problems rather than general quantum computing. They use a technique called quantum annealing to find solutions to complex optimization problems by evolving a quantum system to its lowest energy state.

How It Works:

Qubits in a quantum annealer are initially prepared in a superposition.

The system's Hamiltonian (energy configuration) is gradually adjusted, guiding qubits toward the lowest energy state that represents the optimal solution.

5.2. Technical Challenges in Quantum Hardware

1. Scalability: Today's quantum processors range from a few dozen to a few thousand qubits, but practical quantum advantage likely requires millions.

2. Decoherence: Qubits lose their quantum state due to environmental interference.

3. Error Correction Overhead: Qubits are extremely sensitive to noise. Quantum error correction requires hundreds to thousands of physical qubits per logical qubit.

4. Gate Errors: Imperfect pulses or calibration lead to incorrect quantum logic.

5. Thermal Isolation: Maintaining ultra-low temperatures is expensive and technically challenging. Even micro-watt heat loads can destabilize cryogenic systems.

6. Fabrication Variability: Tiny differences in chip fabrication cause large behavioral shifts in qubit performance.

7. Crosstalk: Operations on one qubit can unintentionally affect nearby ones.

6.Quantum error correction

Quantum error correction (QEC) is a set of techniques used in quantum computing to protect quantum information from errors due to decoherence and other quantum noise. Quantum error correction is theorised as essential to achieve fault tolerant quantum computing that can reduce the effects of noise on stored quantum information, faulty quantum gates, faulty quantum state preparation, and faulty measurements. Effective quantum error correction would allow quantum computers with low qubit fidelity to execute algorithms of higher complexity or greater circuit depth.

Classical error correction often employs redundancy. The simplest albeit inefficient approach is the repetition code. A repetition code stores the desired (logical) information as multiple copies, and—if these copies are later found to disagree due to errors introduced to the system—determines the most likely value for the original data by majority vote. For instance, suppose we copy a bit in the one (on) state three times. Suppose further that noise in the system introduces an error that corrupts the three-bit state so that one of the copied bits becomes zero (off) but the other two remain equal to one. Assuming that errors are independent and occur with some sufficiently low probability p, it is most likely that the error is a single-bit error and the intended message is three bits in the one state. It is possible that a double-bit error occurs and the transmitted message is equal to three zeros, but this outcome is less likely than the above outcome. In this example, the logical information is a single bit in the one state and the physical information are the three duplicate bits. Creating a physical state that represents the logical state is called *encoding* and determining which logical state is encoded in the physical state is called *decoding*. Similar to classical error correction, QEC codes do not always correctly decode logical qubits, but instead reduce the effect of noise on the logical state.

Copying quantum information is not possible due to the no-cloning theorem. This theorem seems to present an obstacle to formulating a theory of quantum error correction. But it is possible to *spread* the (logical) information of one logical qubit onto a highly entangled state of several (physical) qubits. Peter Shor first discovered this method of formulating a *quantum error correcting code* by storing the information of one qubit onto a highly entangled state of nine qubits.

In classical error correction, syndrome decoding is used to diagnose which error was the likely source of corruption on an encoded state. An error can then be reversed by applying a corrective operation based on the syndrome. Quantum error correction also employs syndrome measurements. It performs a multi-qubit measurement that does not disturb the quantum information in the encoded state but retrieves information about the error. Depending on the QEC code used, syndrome measurement can determine the occurrence, location and type of errors. In most QEC codes, the type of error is either a bit flip, or a sign (of the phase) flip, or both (corresponding to the Pauli matrices X, Z, and Y). The measurement of the syndrome has the projective effect of a quantum measurement, so even if the error due to the noise was arbitrary, it can be expressed as a combination of basis operations called the error basis (which is given by the Pauli matrices and the identity). To correct the error, the Pauli operator corresponding to the type of error is used on the corrupted qubit to revert the effect of the error.

7.Application of Quantum Computing

Reading the article so far, you know that "What is Quantum Computing and How it Works". Now let's look at the real-world applications of quantum computing.

Quantum computing can be used in various fields such as:

- Drug Discovery
- Cybersecurity
- Financial modelling and calculations
- Logistics
- Manufacturing, etc.

Let's look at each of the applications in detail.

7.1.Quantum Computing in Drug Discovery

Various application of quantum computing drug discovery includes:

- **Molecular simulation:** To make a simulation of molecular systems, classical computers may face some issues because making this simulations require high computational power but this work can be done easily with a quantum computer. The results provided by quantum computing are highly accurate as compared to classical computing.

Drug Design Optimization: Various quantum algorithms can solve complex optimization problems better than the classical algorithms. This is very important for designing a drug.

7.2.Quantum Computing in Cybersecurity

Quantum computing can be used in both ways as a potential threat as well as a tool for enhancing security.

Quantum Key Distribution (QKD): Secure communication channels can be created by Quantum key distribution by using quantum mechanics.

Secure Multiparty Computing: More security can be added to multiparty computation, where many parties collaborate and compute a function with various inputs and that inputs are kept private.

Quantum Blockchain: The efficiency and security of blockchain technologies can be improved with the help of quantum computing. New ways to make blockchain transactions more secure can be found using quantum hashing algorithms.

7.3. Quantum Computing in Financial Modelling and Calculations

Risk Analysis: To model the probability of different outcomes in financial market, the Monte Carlo simulations are used and it can be enhanced by using quantum computing. These simulations can be performed faster and can get more accurate results with the use of quantum algorithms.

Fraud Detection: Fraud transactions and activities can be monitored more effectively by the use of quantum machine learning algorithms that detects patterns and peculiarity in large datasets.

7.4. Quantum Computing in Logistics

Route Optimization: Quantum computing can optimize the routes for the delivery vehicles and find the best route with which the product can reach the consumer in least possible time and also consume less fuel. This increases the efficiency of goods delivery.

Supply Chain Management: Inventory levels can be optimized across multiple locations with the use of quantum computers and this ensures that the goods are available where needed and this minimizes the storage costs as well.

Warehouse Management: The designing of the warehouse can be done using quantum computing by which the utilization of space can be done in the best possible way and the time required to pick and pack items can also be decreased. The paths of the robots working in the warehouse can be designed by quantum algorithms that ensure efficient and collision free operations.

7.5. Quantum Computing in Manufacturing

Production Process Optimization: The production process can be optimized by finding the best way to use resources, reduce the production time and reduce the waste generation. The quantum algorithms can schedule the production, which makes sure that the workers and machines are used in the most effective way. This scheduling will increase the output and reduce the downtime.

Quality Control and Defect Detection: Quantum algorithms can predict machine / equipment failure before the occurrence of the failure. This can reduce the downtime by enabling proactive maintenance. The detection of defects in the manufactured products can be enhanced as quantum computing can analyze large datasets and find the defects in the products through sensors.

8. Conclusion

Quantum computing is transitioning from theory to reality, with major strides in hardware, software, and algorithm development. While still in its infancy, the potential is enormous. Continued collaboration across academia, industry, and government is key to unlocking its full promise.

CHAPTER – 2 Quantum Computing Technologies: Foundations, Implementations, and Future Directions

Quantum computing technology is a transformative field that harnesses the laws of quantum mechanics—such as superposition, entanglement, and interference—to process information in fundamentally new ways. Unlike classical computers, which use bits (0 or 1), quantum computers use **quantum bits or qubits**, which can represent 0, 1, or both simultaneously.

This allows quantum systems to perform complex computations exponentially faster than their classical counterparts for specific problem classes, such as factoring large numbers, searching unsorted databases, or simulating quantum systems in chemistry and materials science.

1.Introduction

Quantum computing harnesses non-classical phenomena to perform computations more efficiently than classical architectures for certain tasks. Since the proposal of quantum algorithms by Feynman (1982) and Deutsch (1985), researchers have sought to build hardware capable of realizing these algorithms. Unlike classical bits, which exist deterministically as 0 or 1, quantum bits (qubits) can occupy superposed states, enabling massive parallelism. However, qubit fragility and error sensitivity pose significant engineering challenges. This paper surveys the foremost quantum computing technologies, examining their physical principles, current state-of-the-art performance, and prospects for scaling.

2. Fundamental Concepts in Quantum Computing

2.1 Qubits and Superposition

A qubit is a two-level quantum system described by state |ψ⟩=α|0⟩+β|1⟩\lvert\psi\rangle = \alpha\lvert0\rangle + \beta\lvert1\rangle, where |α|2+|β|2=1\lvert\alpha\rvert^2 + \lvert\beta\rvert^2 = 1. Superposition allows a single qubit to represent both basis states simultaneously, forming the basis of quantum parallelism.

2.2 Entanglement and Quantum Gates

Entanglement correlates multiple qubits such that the state of each cannot be described independently. Controlled-NOT (CNOT) and Hadamard (H) gates generate and manipulate entanglement. Quantum gates correspond to unitary operators acting on the Hilbert space of qubits, preserving normalization and enabling reversible computation.

2.3 Quantum Circuits and Algorithms

Quantum circuits arrange gates in sequence to implement algorithms. Landmark examples include Shor's integer factorization and Grover's search. Circuit depth, gate count, and connectivity constraints impact algorithm feasibility on a given hardware.

2.4 Error Correction and Fault Tolerance

Quantum error correction encodes logical qubits into entangled states of multiple physical qubits, detecting and correcting both bit-flip and phase-flip errors without directly measuring the encoded information. Surface codes and color codes represent leading approaches, requiring large physical-to-logical qubit ratios.

3. Criteria for Evaluating Quantum Technologies

To compare platforms, we define key metrics:

Coherence Time (T1, T2): The characteristic time scales for energy relaxation and dephasing, respectively. Longer coherence enables deeper circuits.

Gate Fidelity: The average accuracy of single- and two-qubit operations, measured via randomized benchmarking. High fidelity (>99.9%) is essential for fault-tolerant thresholds.

Scalability: Physical layout and control complexity affecting qubit counts. Includes on-chip integration and crosstalk management.

Control Complexity: The instrumentation (microwave electronics, lasers) required to implement gates and readout.

Environmental Requirements: Temperature, vacuum, and isolation standards necessary to maintain quantum coherence.

4.Overview of Quantum Computing Technologies

4.1 Superconducting Qubits

Principle: Superconducting qubits leverage Josephson junctions embedded in microwave resonators at millikelvin temperatures. The two lowest energy levels define the qubit. Fast microwave pulses enact single- and two-qubit gates via resonant control.

Performance: Modern transmon devices achieve single-qubit fidelities >99.9% and two-qubit fidelities ~99.5%. Coherence times (T1) range from 50 to 200 µs. Qubit counts in commercial systems exceed 100.

Challenges: Cryogenic refrigeration to <20 mK, frequency crowding, and crosstalk limit scaling. Ongoing work targets 3D integration and alternative materials to extend coherence.

4.2 Trapped Ion Qubits

Principle: Single atomic ions (e.g., ^{171}Yb^+) are confined by electromagnetic fields in ultra-high vacuum. Qubit states reside in hyperfine or Zeeman levels. Laser pulses implement gates via stimulated Raman transitions.

Performance: Gate fidelities exceed 99.9% for single qubits and ~99.7% for two-qubit entangling gates. Coherence times can reach seconds or longer. Arrays of 10–100 ions have been demonstrated.

Challenges: Laser complexity, gate speeds ~10–100 µs, and motional mode crosstalk. Scalability efforts include microfabricated surface traps and photonic interconnects.

4.3 Photonic Qubits

Principle: Information is encoded in discrete photon properties—polarization, time bins, or spatial modes. Linear optical elements (beam splitters, phase shifters) and single-photon detectors realize gates probabilistically or via measurement-based approaches.

Performance: Room-temperature operation and long coherence in optical fibers. Gate success rates are probabilistic, demanding multiplexed sources and feed-forward control.

Challenges: Deterministic two-qubit operations remain elusive. On-chip photonic integration and quantum dot single-photon sources are active research areas.

4.4 Spin Qubits in Semiconductors

Principle: Electron or nuclear spin states in quantum dots or donor atoms serve as qubits, manipulated by electric and magnetic fields. Silicon-based fabrication promises compatibility with CMOS technology.

Performance: Recent spin qubits demonstrate single-qubit fidelities >99% and coherence times up to milliseconds using isotopically purified silicon. Two-qubit gates are emerging with fidelities ~98%.

Challenges: Precise fabrication, charge noise, and integration of control lines at scale. Approaches include multiqubit arrays with shared reservoirs and gate stacks.

4.5 Topological Qubits

Principle: Non-Abelian anyons in topological superconductors store quantum information in global system properties, inherently protected from local noise. Logical gates correspond to anyon braiding.

Performance: Still experimental. Signatures of Majorana zero modes have been observed, but robust qubit demonstrations are pending.

Challenges: Material synthesis, quasiparticle poisoning, and measurement of non-Abelian statistics. Long-term potential for low-overhead fault tolerance.

5.Comparative Analysis

Technology	Coherence	1Q Fidelity	2Q Fidelity	Qubit Count	Scalability
Superconducting	50–200 μs	>99.9%	~99.5%	100+	Moderate (3D IC)

Technology	Coherence	1Q Fidelity	2Q Fidelity	Qubit Count	Scalability
Trapped Ions	>1 s	>99.9%	~99.7%	10–100	Moderate (surface traps)
Photonic	>km in fiber	N/A	Probabilistic	N/A	Challenging
Spin Qubits	ms	~99%	~98%	<10	Potentially high
Topological (theory)	>>1 s	Theoretical	Theoretical	—	High (fault-tolerance)

Each platform offers a distinct balance: superconducting and trapped ions lead current scale, while photonic and spin qubits promise specific application niches and integration advantages.

6.Industry Landscape and Progress

IBM and **Google** have demonstrated 100+ qubit superconducting processors, offering cloud access via Qiskit and Cirq, respectively. **IonQ** and **Quantinuum** provide trapped-ion systems with high-fidelity gates. **Xanadu** and **PsiQuantum** pursue photonic approaches, aiming for room-temperature, networked quantum processors. **Microsoft** invests in topological qubits, collaborating with university consortia on material platforms. Government programs (e.g., U.S. National Quantum Initiative) and academic institutions worldwide accelerate foundational research and workforce development.

Real-World Applications of Quantum Computing Technology

Although still in the early stages, quantum computing is already showing promising applications in fields such as:

Pharmaceuticals: Simulating complex molecules for faster drug discovery.

Finance: Optimizing portfolios and risk modeling using quantum algorithms.

Logistics: Solving scheduling and routing problems with quantum annealing.

Machine Learning: Quantum-enhanced models for faster pattern recognition.

Cybersecurity: Developing quantum-resistant encryption protocols.

Climate Modeling: Simulating atmospheric chemistry with higher fidelity.

7.Challenges and Limitations

Decoherence & Noise: Environmental coupling limits coherence; error rates exceed fault-tolerance thresholds in many cases.

Error Correction Overhead: Surface codes require hundreds to thousands of physical qubits per logical qubit, emphasizing the need for higher fidelities.

Scalability: Control wiring, cryogenics, and cross-talk management become increasingly complex beyond 1000 qubits.

Standardization: Diverse hardware and software ecosystems hinder interoperability and benchmarking.

8.Future Directions

Hybrid Architectures: Co-design of classical and quantum processors to optimize variational algorithms and error mitigation.

Materials Advances: Novel superconductors, low-loss dielectrics, and topological materials to extend coherence.

Quantum Networking: Development of repeaters and photonic interconnects enabling distributed quantum computing and secure communications.

Integrated Photonics & 3D Packaging: Monolithic integration of control electronics and qubit arrays to reduce footprint and latency.

Roadmap to Fault Tolerance: Progressive scaling with concatenated codes, logical qubit benchmarks, and demonstration of error-corrected logical operations.

9.Conclusion

Quantum computing technologies are rapidly evolving, with multiple platforms advancing toward greater qubit counts and improved fidelities. While superconducting and trapped-ion systems currently lead in scale, photonic, spin, and topological approaches offer promising paths for specialized applications and fault tolerance. Overcoming decoherence, control complexity, and error correction overhead will be essential to realize practical, large-scale quantum machines. Continued interdisciplinary collaboration across academia, industry, and government will drive the field toward its transformative potential.

C. Quantum Image Processing: A New Paradigm for Visual Computing

Abstract

Quantum image processing (QIP) is a research branch of quantum information and quantum computing. It studies how to take advantage of quantum mechanics' properties to represent images in a quantum computer and then, based on that image format, implement various image operations. Due to the quantum parallel computing derived from quantum state superposition and entanglement, QIP has natural advantages over classical image processing. But some related works misuse the notion of quantum superiority and mislead the research of QIP, which leads to a big controversy. In this paper, after describing this field's research status, we list and analyze the doubts about QIP and argue "quantum image classification and recognition" would be the most significant opportunity to exhibit the real quantum superiority. We present the reasons for this judgment and dwell on the challenges for this opportunity in the era of NISQ (Noisy Intermediate-Scale Quantum).

1.Introduction

Since the concept of quantum computing was proposed by Feynman in 1982 , the achievements by many geniuses have shown that quantum computing has dramatically improved computational efficiency. The theory to implement quantum computing is nearly mature; the challenge of realizing universal quantum computing mainly comes from technical issues, such as manipulating large-scale qubits. In recent years, with the successional breakthroughs in quantum technology, quantum computing has entered the era of NISQ (Noisy Intermediate-Scale Quantum), when it is supposed as ready to display "Quantum Supremacy" in some practical application areas.

Among these applications, quantum image processing (here mainly refers to the image classification and recognition) is most likely to be a killer app in the near future and, for the commercial reason, to be the favorite one of big companies (Google, IBM, Intel, etc.). Some new works have already made considerable progress. For example, in this February, Llyod et al. described the method of training the map from classical data to quantum states (maximizing the gap between mapped classes in Hilbert space), which has the power to distinguish the images of ant and bee. Whether the method is superior to classical image recognition is unknown, but it does show the great potential of quantum image recognition.

Besides image classification and image recognition, quantum image processing (QIP) involves exploiting quantum properties to represent, manipulate, compress, and address other issues related to images in a quantum computer. The quantum properties lead to the higher efficiency of representing and manipulating images in the quantum context but also bringing about troubles that do not exist in the classical environment. For example, when representing images with quantum states, one cannot replicate images since quantum states obey the noncloning principle. For another example, if one wants to obtain the image manipulating result by measurements, he must find a smart way to handle state collapsing.

The missing or improper way to handle these issues in some published papers leads to a big controversy. Some scholars even deem QIP is a "quantum hoax." Honestly, their doubts make sense. Not all classical image operations are worthy of implementing in the quantum realm. Still, we note there are branch research studies in QIP that can maximize quantum superiority and weaken or even eliminate these troubles derived from quantum properties. We argue "quantum image classification and recognition" would be the candidate. We will give the reasons for this judgment after describing the research status of QIP and then discuss the opportunities and challenges in this direction.

2. Research Status of Quantum Image Processing

The study of QIP started in 1997. In the early days, very few scholars paid attention to this direction, and the publications were also very few. In recent years, frequent relevant publications in major journals indicate QIP is heating up. The research topics have become rich and hierarchical, which denotes QIP has become an independent research branch of quantum information and quantum computing. In the following sections, we will discuss QIP from two aspects, quantum image format and quantum image operations.

2.1. Quantum Image Format

Quantum image format is the core topic of QIP. Qubit Lattice, Real Ket, and FRQI are three major quantum image formats.

Qubit Lattice is the first quantum image format proposed by Venegas-Andraca. He said if the frequency value (color value) of the light wave can be mapped to the probability amplitude of a qubit, then the pixel value of i^{th} row and the j^{th} column can be stored in the amplitude angle shown in equation (**1**), and the whole image can be represented as a qubit string (equation (**2**)).

$$\left|\text{pixel}_{i,j}\right\rangle = \cos\frac{\theta_{i,j}}{2}\,|0\rangle + \sin\frac{\theta_{i,j}}{2}\,|1\rangle\ , \quad (1)$$

$$|\text{image}\rangle = \left\{\left|\text{pixel}_{i,j}\right\rangle\right\} \quad i = 1, 2, \ldots, n_1,\ j = 1, 2, \ldots, n_2. \quad (2)$$

This representation scheme's essence is to map the image's spatial information to the amplitude of a single qubit without using quantum properties of superposition and entanglement.

The Flexible Representation of Quantum Images (FRQI) proposed by Le et al. was an upgraded version of Qubit Lattice by exploiting quantum state superposition. The scheme still maps each pixel's grayscale value to the amplitude, meanwhile introducing an auxiliary qubit to denote the spatial position of each pixel. Then, the whole image is prepared into a large quantum superposition state. Equation (4) depicts a $2^n \times 2^n$ quantum image, where i can be regarded as an indicator of pixels' position (row × column converted to a one-dimensional vector). Due to quantum states' superposition effect, the representation (storage) space decreases exponentially compared to the classical image.

$$\left|\text{pixel}_i\right\rangle = \cos\,\theta_i\,|0\rangle + \sin\,\theta_i\,|1\rangle\ , \quad (3)$$

$$|\text{image}\rangle = \frac{1}{2^n}\sum_{i=0}^{2^{2n}-1}\left(\cos\,\theta_i\,|0\rangle + \sin\,\theta_i\,|1\rangle\right)\otimes|i\rangle \quad \theta_i \in \left[0, \frac{\pi}{2}\right]. \quad (4)$$

Due to the merit (small storage space, simple, and easy to understand) of FRQI, many follow-up studies have been carried out to extend the scheme. Instead of mapping the pixel's gray value to the amplitude angle, Zhang et al. used a group of ground states to represent the pixel's value (a qubit string) in a larger Hilbert space. This scheme facilitates some image operations and improves efficiency. Wang et al. extended FRQI to polar coordinates and replaced pixels' spatial position with polar diameters and polar angles. Ruan et al. expressed the pixel's gray value as $|0\rangle + e^{i\vartheta}|1\rangle$, replacing the $\cos\vartheta|0\rangle + \sin\vartheta|1\rangle$ that stored the gray value in FRQI, which actually replaced the ZOX plane's rotation on Bloch ball with a rotation of ϑ angle in the XOY plane. All these extended schemes' basic idea is to prepare the image into a quantum superposition state in terms of pixels' spatial distribution information, which is not fundamentally different from FRQI.

The last quantum image format, Real Ket, was proposed by Latorre. He divided the image into 2 × 2 pieces and then mapped the four pixels' grayscale value to the probability amplitude of each component of a quantum state with 2 qubits. Equation (5) describes this quantum state, where $i_1 = 1$ can be understood as the index of the top-left pixel, $i_1 = 2$ as the index of the top-right pixel, $i_1 = 3$ as the bottom-left pixel, and $i_1 = 4$ as the bottom-right pixel. C_{i_1} stores the mapping value of each pixel and satisfies $\sum_{i_1=1,\ldots,4}|C_{i_1}|^2 = 1$.

$$|\psi_{2^1\times 2^1}\rangle = \sum_{i_1=1,\ldots,4} C_{i_1}\,|i_1\rangle \quad \text{st.}\sum_{i_1=1,\ldots,4} \left|C_{i_1}\right|^2 = 1. \tag{5}$$

In Ref. , this normalized quantum state is called qudit. If one rewrites it in terms of qubits, a qudit is actually a superposition state of 2 qubits.

Expanding this 2 × 2 block once, one can obtain a 4 × 4 block, which can be represented as a quantum state shown in equation (6), where $i_2 = 1$ can be regarded as the index of the first (upper left corner) 2 × 2 small block, $i_2 = 2$ is the index of the second (upper right corner) 2 × 2 small block, $i_2 = 3$ is the index of the third (lower left) 2 × 2 small block, and $i_2 = 4$ is the index of the fourth (lower right) 2 × 2 small block. In the process of construction, the quantum state must be normalized again, i.e., satisfies

$$\sum_{i_2,i_1=1,\ldots,4} |C_{i_2,i_1}|^2 = 1.$$

$$|\psi_{2^2\times 2^2}\rangle = \sum_{i_2,i_1=1,\ldots,4} C_{i_2,i_1}\,|i_2 i_1\rangle \quad \text{st.} \sum_{i_2,i_1=1,\ldots,4} \left|C_{i_2,i_1}\right|^2 = 1. \tag{6}$$

Through expanding increasingly, a $2^n \times 2^n$ image can be mapped to the quantum state as shown in the following equation:

$$|image\rangle = |\psi_{2^n\times 2^n}\rangle = \sum_{i_n,\ldots,i_1=1,\ldots,4} C_{i_n,\ldots,i_1}\,|i_n,\ldots,i_1\rangle \ \text{st.} \sum_{i_n,\ldots,i_1=1,\ldots,4} \left|C_{i_n,\ldots,i_1}\right|^2 = 1. \tag{7}$$

The basic idea of this representation is using the basis to represent the pixel's spatial position while using the probability amplitude to represent the color information. Ref. exploit this idea to implement similar schemes.

Let us analyze the storage efficiency of these formats briefly. For a $2^n \times 2^n$ gray image, if the gray value of each pixel is represented by 8 classical bits, then a total of $2^n \times 2^n \times 8$ bits are required for the classical image. As for the Qubit Lattice, only one qubit is needed to represent a single pixel's grayscale value, so $2^n \times 2^n$ qubits are needed for the whole image. In FRQI, a qubit's gray value is still represented by a qubit, but since the whole image is prepared into a superposition state in terms of the row and column coordinates, only $2n + 1$ qubits are needed. In the last format Real Ket, as the pixels' grayscale information is stored in the probability amplitude of the components of a superposition quantum state, the storage space only need $2n$ qubits. Thus, Real Ket uses minimum storage space.

2.2. Quantum Image Operation

Image operations involve a wide variety of types. In recent years, many works have discussed this topic in the quantum realm, and the operations realized are increasingly abundant. Here, we roughly classify these operations into geometric transformation, color transformation, and complicated operations such as compression and retrieval. We describe them, respectively, as follows.

2.2.1. Geometric Transformation

Geometric transformation refers to changing the spatial position (coordinate) of pixels, such as image rotation, or changing image shape, like zoom in or out. The realized operations include rotation, the top/bottom swapping or left/right swapping, the interchange of any two coordinates of the image, erosion and dilation operations, image magnification, and overall translation and cyclic translation of the image.

Take FRQI as an example. Its format can be simplified as qubits representing the color information tensor by qubits representing pixels' coordinate information (equation (8)). k is the position component, representing N pixel's spatial position. C_k is the color component, representing the k^{th} pixel's gray value.

$$\left|\text{image}\right\rangle = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}\left|C_k\right\rangle \otimes \left|k\right\rangle . \quad (8)$$

In general, geometric transformation refers to realizing the image operation operator acting on the position component. According to the scope of the operator, we can classify it into global operation and local operation.

As the name implies, global operations operate on all pixels of the image, such as rotation (rotate 90°, 180°, 270°). Equation (9) gives a formal description of such operations, where G is the unitary transformation acting on all pixels.

$$G\left(\left|\text{image}\right\rangle\right) = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}\left|C_k\right\rangle \otimes G\left(\left|k\right\rangle\right) . \quad (9)$$

Since each pixel's gray values are stored in the corresponding component of the superposition state, performing a unitary operation *G* would modify all components simultaneously (quantum parallelism). Thus, global operation *G*'s efficiency is much higher than the counterpart operation to a classical image.

Local operations only involve the manipulation of a few pixels, leaving other pixels unchanged. For example, swap the coordinate of i^{th} and j^{th} pixels. Then, there should be a unitary operator *S* that satisfies

$$S\left(\left|\text{image}\right\rangle\right) = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}\left|C_k\right\rangle \otimes S\left(|k\rangle\right),$$

$$\text{Where } S\left(|k\rangle\right) = |k\rangle\,, k \neq i, j \text{ and } S\left(|i\rangle\right) = \left|j\right\rangle, S\left(\left|j\right\rangle\right) = |i\rangle\,. \quad (10)$$

The operator *S* can be constructed as equation (**11**). Swapping two pixels only needs this operator to act once.

$$S = |i\rangle\left\langle j\right| + \left|j\right\rangle\langle i| + \sum_{k\neq i,j}|k\rangle\langle k|\,. \quad (11)$$

Consider the counterpart operation in a classical image. Swapping two pixels is equivalent to switching *i* and *j* in an array. That means one needs to perform at least two writes, one at *i* and one at *j*. Obviously, local quantum operations are also faster than classical processes.

2.2.2. Color Transformation

Color transformation refers to changing the image's pixel value, such as halftone processing of images. If such changes can be restored by some means, then image watermark, shuffling color blocks, image encryption/decryption, etc., can be regarded as this category.

In terms of the acting scope of color transformation, one can also divide it into global operation and local operation. Using a similar analysis above, one can see that the global color transformation achieves exponential acceleration compared to the classical counterpart and local operation is faster than its classical counterpart. Because the quantum Fourier transform is invertible and exponentially faster than the classical Fourier transform, it is widely used in image watermarking and image encryption/decryption. Due to the "uncertainty" principle of quantum mechanics, quantum image watermarking, quantum image encryption/decryption, etc., can guarantee hidden information security.

2.2.3. Complicated Image Operations

In classical image processing, effective methods to achieve compression, retrieval, recognition, segmentation, registration, and other operations generally need to perform some preprocessing based on the original image data, such as transforming the image domain or extracting image features. In this paper, we call these operations as “complicated” image operations.

Compression is the most discussed topic. In Ref. , quantum image compression is defined as reducing the number of quantum gates when preparing quantum images (quantum states). This definition is different from classical image compression, and in essence, the discussion of such problems can be defined as the optimization of quantum circuits. Ref. describe the methods of performing such compression. Although these methods are slightly different, the basic idea is to reduce the number of quantum gates by simplifying Boolean expressions. A more natural definition of compression would be how to reduce the number of qubits representing a quantum image (corresponding to classical image compression), that is, to reduce the dimensions of the Hilbert space representing the quantum state (quantum image). From the perspective of information theory, it seeks to represent a quantum image with more concentrated energy, i.e., concentrate image primary information into a smaller dimension. Ref. proposed a transforming method based on the matrix product state theory. And this mathematical form makes it possible to obtain the optimum lossless compression and enables us to distinguish the important information and redundant information to perform lossy compression.

Another frequently discussed topic is image retrieval. In quantum image processing, retrieval has two meanings: one is to retrieve classical information from a quantum state and the other is to retrieve the quantum information in the quantum state. The former’s implemental method has been described in Ref.. With the same idea, a large number of quantum images (quantum states) are prepared, the probability amplitude of each ground state is estimated by measurements repeatedly, and then the original quantum image (quantum states) is recovered according to the probability distribution. The latter is based on the retrieval of quantum image content. Schutzhold presented a quantum algorithm for finding simple patterns (such as a parallel line) in black-and-white binary images. The algorithm utilizes quantum Fourier transform characteristics to work in parallel and can achieve exponential speedup compared to classical algorithms. Venegas-Andraca described the relationships between the vertices of graphs such as triangles and squares with quantum entanglement and exploited Bell inequality to provide a method to retrieve the existence of these graphs in black-and-white binary images.

Besides, Le et al. discussed image segmentation. The algorithm uses the operator prepared by an orthogonal basis and the gray level information encoded into the ground state of the quantum state (orthogonal basis) to make an equivalent determination and uses Grover algorithm to accelerate this process. Caraiman realized image segmentation based on the threshold by calculating histogram. Zhang et al. discussed image registration by giving an ordinal number to images with different rotation angles and then using the Grover algorithm to retrieve the ordinal number. This approach is similar to binding a keyword to each image and then retrieving it based on the keyword, rather than content-based retrieval.

2.3. Discussion

Applying the quantum properties of superposition and entanglement to map classical images and store them in qubits is the basic idea of preparing quantum images. Due to the parallel computing induced by the quantum superposition effect, the quantum image operation's efficiency is much higher than the corresponding classical image operation. But if taking into account the cost of quantum image preparation and the cost of obtaining the image manipulation result by measurements, the claim that exponential acceleration of quantum image operation may not exist. Mastriani enumerated these doubts (similar challenges in quantum machine learning) and concluded that many published works related to QIP are "Quantum Hoax." His main viewpoints can be summarized as follows:

(1) Input: many published papers did not consider/reckon the cost of preparing quantum states (images) from classical data (image).

(2) Output: obtaining the result of image operation requires an exponential scale of measurements.

(3) Noise: quantum image is sensitive to noise and simulation software such as MATLAB is not capable of verifying the correctness of quantum algorithms.

His criticism triggered a fierce debate. This March, the journal Quantum Information Processing published both Li et al.'s comments on Mastriani's original paper and Mastriani's rebuttal to these comments. The opinions from both sides partly make sense, but neither seems to be quite right.

For the first doubt, FRQI or other image formats are just a method to map classical data to quantum data without theoretical defects. These quantum images can be efficiently prepared. The interested reader can refer to Yao et al.'s work for a detailed discussion on this issue.

For the second and third doubts, if image operations like geometric transformation and color transformation are one's final goals, he has to measure each pixel to get the results. That procedure for a quantum state (quantum image) is named quantum state tomography, which requires an exponential scale of measurements for a general state. Besides, since the measurement result is the statistical result of the observed value, it is difficult to eliminate the measurement noise. Thus, in quantum image processing, this kind of research work would have little practical significance.

But, if one only wants to take advantage of the quantum image's overall characteristics or some statistical properties rather than read all pixels' value. Quantum image operations like geometric transform and color transform are intermediate steps to the end. These operations would make sense. For example, the HHL algorithm for solving linear equations presents the solution hidden in the quantum superposition state. The correct way to use this solution is to exploit this state's overall property rather than measure it to get the probability distribution for each component. This approach to using HHL forms the foundation of many quantum machine learning algorithms.

Whether the simulation software MATLAB can verify the algorithm of quantum image processing should be considered from two aspects. First, quantum image operations are certain quantum algorithms which can be described by sequential unitary transformations acting on a complex vector in a Hilbert space. Certainly, MATLAB has the capability to simulate such unitary evolution. Second, to our knowledge, there seems to be no module in MATLAB that can simulate quantum noise. Therefore, if one wants to verify quantum algorithms' performance on real quantum computing devices, IBM's qiskit or other tools would be a better choice.

3. Opportunities and Challenges

3.1. Opportunities

Quantum algorithms can improve computational efficiency, but it may not apply to all application scenarios. The successful quantum algorithms all have a distinct characteristic: the intermediate procedure of computation is complicated, suitable for constructing quantum state entanglement and superposition for parallel acceleration; the result is simple and often a decisive answer. In that case, entanglement/superposition degenerates at the end of the quantum algorithm, and the probability amplitude of a single basis is 1 or close to 1. For quantum image processing, the task of image classification and recognition accords with this characteristic. The intermediate algorithm execution procedures involve feature extraction, classifier training, and various distance computing. The result only needs to answer "yes or no" to determine whether the image belongs to a specific category.

Moreover, a wide variety of machine learning protocols operate by performing matrix operations on vectors in a high-dimensional vector space. Quantum mechanics is all about matrix operations on vectors in high-dimensional vector spaces. Thus, performing machine learning tasks in the quantum realm would probably be beneficial if one can take advantage of the natural connection between these two disciplines. This path has proved to be correct. Quantum machine learning has made a lot of achievements in recent years. Thus, quantum machine learning algorithms could act as the building blocks for the classification and recognition of quantum images. Here, we would like to introduce two commonly used techniques “swap test” and “inversion test” to show the advantages of these methods.

Figure **1(a)** illustrates the principle of “swap test.” $|x\rangle$ and $|y\rangle$ are two quantum states (quantum images). Using the circuit, one can estimate the similarity between $|x\rangle$ and $|y\rangle$ by measuring whether auxiliary qubit ends up in $|0\rangle$ alone.

$$P(|0_{\text{anc}}\rangle) = \frac{1}{2} + \frac{1}{2}|\langle x|y\rangle|^2 . \quad (12)$$

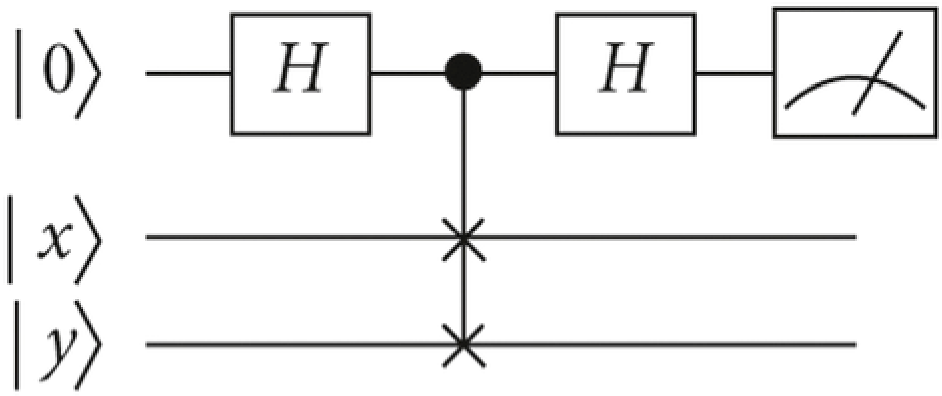


Figure 1 (a)

Quantum computing techniques for distance estimation. (a) Swap test. (b) Inversion test.

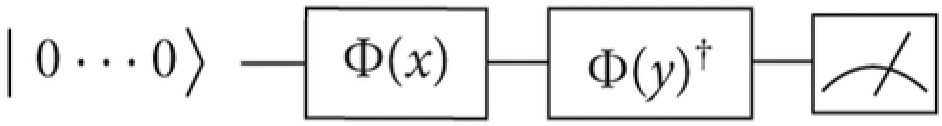


Figure 1 (b)

Quantum computing techniques for distance estimation. (a) Swap test. (b) Inversion test.

$|\langle x|y\rangle|$ is known as “fidelity” in quantum information and “cosine distance” in classical machine learning. One can see that if $|x\rangle$ and $|y\rangle$ are farthest away from each other (orthogonal), this probability is 1/2; if $|x\rangle$ and $|y\rangle$ are closest to each other, this probability is 1. One can also see that this probability estimation has nothing to do with the feature space’s dimension. The higher the dimension is, the more efficient this technique is against the classical algorithm. Lloyd points out that even considering the cost of preparing quantum states, this technique is much more efficient than distance calculations on classical computers. Derived from this technique, many more general distances such as *Euclidean* *distance* = $\sqrt{2 - 2|\langle x|y\rangle|}$ can also be calculated. This technique is widely used in various quantum machine learning algorithms.

The circuit in Figure **1(b)**, proposed by Havlíček et al. in 2019, is known as the "inversion test" to calculate the distance between quantum states. *x* and *y* are features of the classical data (image), as an argument to input into quantum circuit Φ and its inverse circuit $\Phi^{\dagger}$. One can see that if *x* and *y* are equal (similar), the probability should eventually be 1 or near to be 1 when doing the projection measurement on $|0 \ldots 00\rangle$. If *x* and *y* differ a lot, this probability is smaller. This technique can also calculate distances in quantum space (Hilbert space), and its advantage versus "swap test" is reducing a half number of qubits used.

3.2. Challenges

Based on the above discussion, one could conclude that quantum image classification and recognition is the most significant opportunity in quantum image processing. But we must note that the certification of its ability beyond the classical image recognition method in both theory and practice still needs to face the following challenges.

3.2.1. Preparation of Quantum Images

Only by mapping and preparing classical data to quantum superposition state can the advantage of quantum computation be brought into play. The general method to prepare a quantum superposition state is to use QRAM (Quantum Random Access Memory). Its basic idea is to use a "bucket brigade" structure to distribute *N* d-dimensional vector data on the *Nd* leaf nodes of the "tree." Based on this structure, QRAM can prepare *N* d-dimensional vectors into log(*Nd*) qubits superposition state in O(log(*Nd*)) time. However, this structure requires O(*Nd*) physical resources. Its scale is exponential in the number of qubits, so whether it can provide real computational advantages in the actual experimental environment is still a big question.

3.2.2. Feature Extraction

Feature extraction plays a vital role in classical image recognition. Effective features can eliminate the influence of image background, size, lighting conditions, camera angle, etc., and improve image recognition accuracy; however, little literature talks about how to extract features from the mainstream quantum image formats. The absence of discussion on this issue cast doubt on some impressive work. For example, Ref. used Hadamard transform to calculate the difference between adjacent pixels quantum image and then compared the known pattern by "swap test" to detect the image edge. However, if the image pattern has a certain degree of deformation, such as size inconsistency, which will lead to a considerable difference between the known pattern and the quantum image to be tested, is it still valid to use "swap test" to detect the image edge? Moreover, this paper only experiments for image edge detection on binary images; it is unknown that the proposed approach for natural images' detection is still effective.

3.2.3. Nonlinear Operations

Effective procedures in a classical image (pattern) recognition and machine learning tend to be nonlinear, such as the sigmoid function used by training perceptron in deep learning. The nature of quantum mechanics is linear. A feasible scheme is to induce nonlinear operation through measurements. However, the quantum state will collapse after measurements, making the system's evolution lose its quantum characteristics and degenerate into the classical probability perceptron. Another popular solution is to implement nonlinear operation (such as feature extraction) by a classical process and then compute the kernel function in quantum space (Hilbert space). Llyod called this approach as "quantum embedding". This strategy's essence is to circumvent the nonlinear operation in quantum space; whether it is superior to the classical machine learning algorithm needs further verification.

3.2.4. Noise

Noise is one of the most important theoretical problems in quantum computation. The existence of quantum noise may lead to quantum computation to classical probabilistic computation. Since the 1990s, people have been studying quantum error correction codes and further developing the concept of fault-tolerant quantum computing. In recent years, eliminating errors caused by noise has been a fascinating research direction, among which topological quantum computing has aroused the most concern because it promises to solve quantum noise completely. The more interesting thing is that functions that cannot be learned in a noisy classical environment can be learned in the noisy quantum environment, such as Disjunctive Normal Form (DNF). In recent years, this research has extended to more general linear functions such as odd and even functions.

4. Conclusions

Compared with classical image processing, quantum image processing is far from sufficient in both depth and width. The "quantum advantage" claimed in some related published papers has also been doubted by many scholars; the core of these doubts is "how to obtain quantum image operating results efficiently and accurately." We deem that these studies try to get quantum image operation results by recovering classical images via measurements without practical significance.

On the contrary, the kind of research that only exploits quantum images' statistical characteristics may be valuable. With the assistance of quantum machine learning, quantum image classification and recognition would have the most significant opportunity to be a "killer app" in the actual commercial field in the NISQ era.

We are sure that not all classical image manipulation is necessary to implement in a quantum computer. Whether image geometric transformation, color transformation, and other similar operations are worthy of implementing in the quantum realm depends on the application scenario. For example, Yao et al. calculated the difference between adjacent pixels in the quantum image via Hadamard transform (which can be regarded as a kind of color transform) and then used the result directly to obtain the image edge by matching a known pattern. That is, these kinds of quantum image operations make sense, only proving that they are intermediate steps to the end rather than the final goal.

D. Quantum Image Classification: A Comprehensive Study

Abstract

Nowadays, using machine learning for image classification is very common. However, due to the increasing demand for data processing and fast computing, the idea of enhancing machine learning with quantum computing has been proposed, known as quantum machine learning (QML). Quantum machine learning has the advantages of higher efficiency and accuracy. Quantum computing uses quantum bits (qubits) for data storage and computing, where a qubit can represent quantum states $|0\rangle$ and $|1\rangle$ simultaneously, enabling the processing of information for two states simultaneously, which is unparalleled in classical computing. Moreover, quantum machine learning can handle more complex data and process data faster. In classical machine learning, the processing of large-scale data and complex problems often faces problems of high computational complexity and low algorithm efficiency. Quantum computing can handle multiple computing tasks simultaneously, achieving faster computing. Therefore, in some scenarios that require efficient computing, quantum machine learning may be the best choice. In this study, we simulated quantum circuits using Qiskit and built a hybrid quantum-classical neural network model using VQNet to classify MNIST handwritten digits and CIFAR-10 datasets. The experiments showed that quantum machine learning has the advantages of efficiency, accuracy, and security over classical machine learning, which may be an improvement over classical machine learning. This research proposes a machine learning algorithm based on quantum computing, which promotes the development of quantum computing and quantum technology. At the same time, it provides a new solution and idea for image classification, enabling people to pursue faster and more accurate quantum machine learning instead of being limited to classical machine learning.

1.Introduction

Image classification, a pivotal task in multiple industries, faces computational challenges due to the burgeoning volume of visual data. This research addresses these challenges by introducing two quantum machine learning models that leverage the principles of quantum mechanics for effective computations. Our first model, a hybrid quantum neural network with parallel quantum circuits, enables the execution of computations even in the noisy intermediate-scale quantum era, where circuits with a large number of qubits are currently infeasible. This model demonstrated a record-breaking classification accuracy of 99.21% on the full MNIST dataset, surpassing the performance of known quantum-classical models, while having eight times fewer parameters than its classical counterpart. Also, the results of testing this hybrid model on a Medical MNIST (classification accuracy over 99%), and on CIFAR-10 (classification accuracy over 82%), can serve as evidence of the generalizability of the model and highlights the efficiency of quantum layers in distinguishing common features of input data. Our second model introduces a hybrid quantum neural network with a Quanvolutional layer, reducing image resolution via a convolution process. The model matches the performance of its classical counterpart, having four times fewer trainable parameters, and outperforms a classical model with equal weight parameters. These models represent advancements in quantum machine learning research and illuminate the path towards more accurate image classification systems.

3. Quantum Image Representation Models

3.1 Flexible Representation of Quantum Images (FRQI)

FRQI encodes grayscale images by embedding pixel intensity as angle information: $|I\rangle = \frac{1}{2^n} \sum_{i=0}^{2^n-1} (\cos\theta_i |0\rangle + \sin\theta_i |1\rangle) |i\rangle$

Pros: Efficient superposition, simple encoding Cons: Precision limited by angle resolution.

3.2 Novel Enhanced Quantum Representation (NEQR)

NEQR stores pixel intensities in binary across qubits: $|I\rangle = \frac{1}{2^n} \sum_{i=0}^{2^n-1} |f(i)\rangle |i\rangle$

Pros: Precise encoding, easier extraction Cons: More qubits required

3.3 Other Representations

QPIE (Quantum Polar Image Encoding): Circular image encoding

2D-QSNA: Spatial domain-based model with enhanced fidelity

4. Quantum Classification Algorithms

4.1 Variational Quantum Classifiers (VQC)

VQCs use parameterized quantum circuits (PQC) trained via classical optimization. Steps:

1. Encode input features as quantum states
2. Apply PQC with tunable parameters
3. Measure outputs and compute loss
4. Use gradient-based optimizers to update parameters

Advantages: Flexible, NISQ-compatible Limitations: Requires hybrid quantum-classical setup

4.2 Quantum Support Vector Machines (QSVM)

QSVMs map inputs into a high-dimensional Hilbert space using quantum kernels. Classification is performed by computing kernel values on a quantum device.

Advantages: Effective for non-linear problems Limitations: Kernel design complexity, sensitive to noise.

4.3 Quantum Convolutional Neural Networks (QCNN)

QCNNs mimic classical CNNs with quantum gates arranged in convolutional and pooling layers. Feature extraction is performed hierarchically.

Advantages: Suited for image data, efficient Limitations: Hardware intensive, limited depth.

5.Experimental Evaluation

5.1 Datasets

Due to hardware constraints, most QIC experiments use simplified datasets:

MNIST (downsampled)

Fashion-MNIST

Binary classification subsets

5.2 Performance Metrics

Accuracy

F1-Score

Quantum circuit depth and width

Inference time

5.3 Results

Small-scale tests using IBM Quantum Experience show:

VQC achieves 85-92% accuracy on small images

QSVM shows competitive performance to classical SVM

Hybrid models outperform pure quantum in practical settings

6.Comparative Analysis

Method	Accuracy	Speed	Scalability	Hardware Req.
CNN (Classical)	98%	Medium	High	CPU/GPU
VQC	92%	Fast (small n)	Medium	Quantum+Classical
QSVM	90%	Fast	Medium	Quantum
QCNN	85%	Fast	Low	Quantum (High)

7.Implementation Platforms

IBM Qiskit: Python-based SDK for quantum programming

PennyLane: Hybrid quantum ML library with PyTorch/TensorFlow support

Cirq: Google’s framework for quantum circuits

8.Challenges

Noise and Decoherence: Affects accuracy and stability

Qubit Limitations: Current devices have <100 usable qubits

Encoding Overhead: Data loading remains inefficient

Lack of Large Datasets: Limits training and benchmarking

9.Future Research Directions

Error Mitigation and Correction: Essential for reliable QIC

Advanced Representations: More compact and robust encoding schemes

Quantum Transfer Learning: Adapting pre-trained quantum models

Quantum Edge Devices: Deployment on portable quantum hardware

Integration with Classical AI: Hybrid architectures with shared learning

10.Conclusion

Quantum image classification is a promising yet evolving field that seeks to leverage quantum computational power to tackle image recognition challenges. Early experiments show competitive performance on small-scale problems, with potential scalability as hardware matures. Further research is needed in areas of representation, algorithm design, and system integration to realize full advantages.

E. Hybrid Quantum Neural Network

Abstract

This research presents a quantum machine learning models to tackle the growing computational demands of image classification. Hybrid quantum neural network with parallel quantum circuits, designed to operate efficiently in the noisy intermediate-scale quantum (NISQ) era. It achieves a **record-breaking 99.21% accuracy on the full MNIST dataset**, outperforming existing quantum–classical models while using **8x fewer parameters** than its classical counterpart. It also shows strong generalizability with **>99% accuracy on Medical MNIST** and **>82% accuracy on CIFAR-10**, highlighting the effectiveness of quantum layers in feature extraction.

classification.

1.Model

The **HQNN-Parallel** model consists of two key components: a **classical convolutional block** for dimensionality reduction and a core of **classical fully connected layers** combined with **parallel quantum dense layers (PQCs)**. The classical block reduces input data size, while the hybrid core is responsible for the prediction tasks. This architecture efficiently processes data and makes accurate predictions using both classical and quantum layers.

1.1.Classical convolutional block

The general structure of the classical convolutional part of the proposed HQNN-Parallel. The convolutional part of the network is comprised of two main blocks, followed by fully-connected layers. In this study, we utilized Rectified Linear Unit (ReLU) as the activation function. Batch Normalization is employed in the network as it stabilizes the training process and improves the accuracy of the model.

First Convolutional Block:

Input Channels: 1, Output Channels: 16, Kernel Size: 5 × 5, Stride: 1 pixel, Padding: 2 pixels

Output Size: Maintains original input size of 28 × 28

Purpose: Preserve spatial dimensions while extracting complex features

Post-processing: Batch Normalization, ReLU activation, MaxPooling with 2 × 2 kernel

Resulting Feature Map: 16 × 14 × 14

Second Convolutional Block:

Input Channels: 16, Output Channels: 32, Kernel Size: 5 × 5, Padding: 2 pixels, Stride: 1 pixel

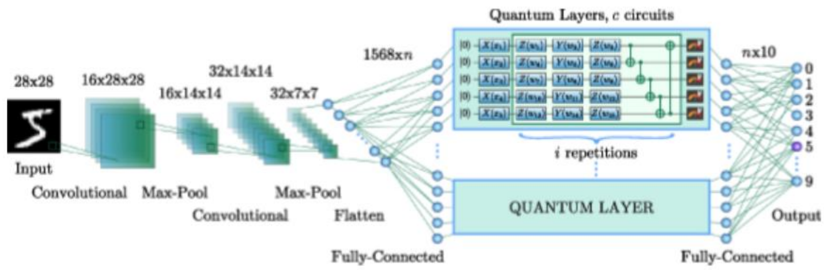
MaxPooling: Same as before

Resulting Feature Map: 32 × 7 × 7

This final output is passed to the fully connected (dense) part of the network.

1.2. Hybrid Dense Layers:

After the convolutional layers, the HQNN-Parallel architecture moves into the hybrid dense section.



Zoom In Zoom Out Reset image size

Figure 4. Architecture of the proposed HQNN-Parallel. The input data samples are transformed by a series of convolutional layers, that extract relevant features and reduce the dimensionality of the input. The output channels of the convolutional layers are then flattened into a single vector before being fed into the dense part of the HQNN-Parallel. The hybrid dense part contains a combination of classical and quantum layers. The quantum layers are implemented using parallel PQCs, which allow for simultaneous execution, reducing the total computation time. Quantum layers are depicted in the figure as blue rectangles, the top rectangle is a detailed version of subsequent quantum layers in the amount of *c* circuits. The output of the last classical fully connected layer is a predicted digit in the range of 0–9.

Input to Dense Layer:

The final output from the convolutional block is a feature map of size 32 × 7 × 7.

This is flattened into a vector of size 1568 (i.e., 32 × 7 × 7 = 1568).

This vector becomes input to the first dense layer.

Transformation:

The first dense layer transforms the 1568 features into n features.

The value of n is defined by the quantum component, representing the total number of encoding parameters in the quantum layers.

Quantum Layer Function:

Each quantum layer:

Maintains same number of input and output features (preserves n).

Its output is fed into the second classical fully connected layer.

This final dense layer maps n features to 10 output features, where 10 is the number of classes for classification.

Post-Processing:

After each classical dense layer, the network applies: Batch Normalization, ReLU Activation

Flexibility:

The structure of HQNN-Parallel, including:

The number of layers

The number of features

Can be tuned or adjusted to improve task-specific performance.

1.3. Structure of Quantum Layer:

The quantum component has c parallel quantum layers.

Each quantum layer is a Parameterized Quantum Circuit (PQC) with 3 parts:

- Embedding
- Variational gates
- Measurement

Input to Quantum Layers:

Input from the previous dense layer: n features

This input is split into $c = n / q$ parts, each part being a vector of q values.

1. Embedding:

Uses angle embedding: rotates qubits around X-axis of the Bloch sphere based on input values.

Encoding formula:

$|\psi\rangle = R_x^{emb}(x)\ |0\rangle^n$

This encodes classical data into quantum state.

2. Variational Part:

Each PQC has a rotation layer (with trainable parameters) followed by CNOT operations.

These act as quantum gates that:

Adjust states based on learnable parameters.

Introduce entanglement between qubits.

The number of repetitions (i) of this structure is a hyperparameter.

3. Measurement:

After operations, Pauli-Y basis measurement is done:

Resulting in a value:

$v_j = \langle 0|\ R_x^{emb\dagger}\ Y_j\ U^\dagger(\theta)\ U(\theta)\ R_x^{emb}\ |0\rangle$

This gives q-dimensional vectors for each PQC.

Final output: all vectors are concatenated into $v \in \mathbb{R}^n$, serving as input for the next classical dense layer.

Final Classification:

The final dense layer outputs a probability distribution over 10 classes (digits 0–9).

Softmax-style output: The neuron with the highest probability is the predicted class.

Weight Count:

Total trainable parameters in quantum part:

$q \times 3 \times i \times c$

1.4. Test and Result

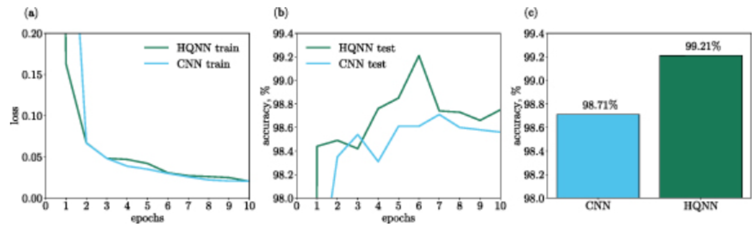
As described above, the HQNN-Parallel was trained on MNIST dataset section 2.1.1. No preprocessing is applied, so the entire collection is used for training (60000 images are in the training set and 10000 are in the test set). In the context of training the proposed HQNN-Parallel, the ultimate objective is to minimize the loss function during the optimization process. The cross-entropy function is employed as the loss function, given by:

$$l = -\sum_{c=1}^{k} y_c \log p_c,$$

where p_c is the prediction probability, y_c is either 0 or 1, determining respectively if the image belongs to the prediction class, and k is the number of classes.

The **classical layers** in the HQNN are optimized using **backpropagation**, implemented through the **PyTorch** library, which calculates gradients via automatic differentiation. In contrast, **quantum layers** require a different approach. For this, we use the **PennyLane** framework and apply the **parameter shift rule**, a technique suitable for quantum hardware. This method computes gradients by slightly shifting quantum circuit parameters and measuring output changes, allowing effective training of the **variational quantum circuits** and enabling the HQNN to learn complex patterns efficiently.

During experimentation, we tested various quantum architectures. The most effective used **5 qubits** and **3 repetitions** of **strongly entangling layers**, repeated across **4 quantum layers**. This **HQNN-Parallel** achieved **99.21% accuracy** on the MNIST dataset. For comparison, we replaced the quantum layers with classical dense layers (same convolutional part) and trained the modified CNN on the same data. The HQNN outperformed its classical counterpart, as shown in the results:



Zoom In Zoom Out Reset image size

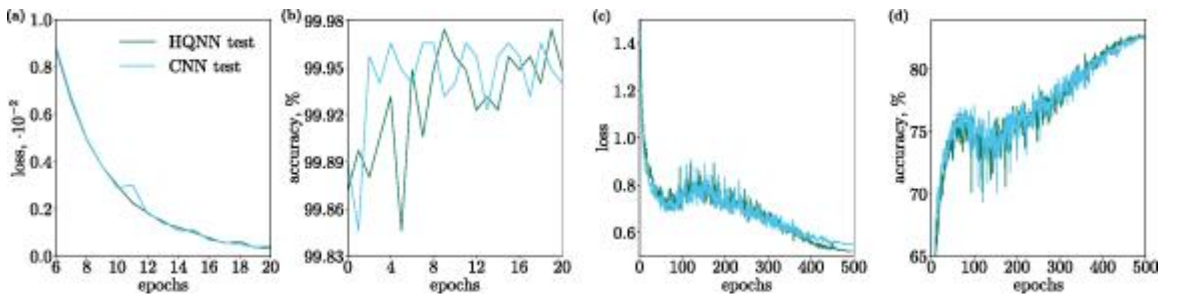
Figure 5. (a) and (b) Train and test results for the HQNN-Parallel and the CNN. The HQNN has a 99.21% accuracy on the test data and outperforms the CNN which has a 98.71% accuracy. The classical model has 8 times more variational parameters than the hybrid one. (c) Test accuracies of the HQNN-Parallel and its classical counterpart, the CNN.

The trainable parameters, as well as the primary training and testing results, for both the HQNN-Parallel and the CNN, are summarized in table 1 and illustrated in figure(c). From these results, it is evident that the most successful implementation of the HQNN-Parallel surpasses the performance of a CNN that possesses approximately eight times more parameters.

Table 1. Summary of the results for the HQNN-Parallel and its classical analog, CNN.

Dataset	Model	train loss	test loss	test acc	param num
MNIST	CNN	0.0205	0.0449	98.71	372 234
	HQNN	0.0204	0.0274	99.21	45 194
Medical	CNN	$0.456 \cdot 10^{-2}$	$0.396 \cdot 10^{-2}$	82.64	247 642
MNIST	HQNN	$0.429 \cdot 10^{-2}$	$0.332 \cdot 10^{-2}$	82.78	247 462
CIFAR-10	CNN	0.3659	0.5484	82.78	81 698
	HQNN	0.3851	0.5208	82.64	81 578

HQNN-Parallel with its classical analog were also tested on Medical MNIST dataset. The HQNN-Parallel managed to achieve a 99.97% accuracy. The classical CNN showed less accurate results with 99.96% accuracy on test data. It is worth noting that HQNN model had 247 462 trainable parameters and classical CNN had 247 642. A comparison of the training outcomes is depicted in figures (a) and (b).



To test generalizability, we evaluated both **HQNN-Parallel** and classical CNN models on the **CIFAR-10** dataset. The HQNN achieved **82.78% accuracy**, slightly outperforming the classical model's **82.64%**, despite using fewer trainable parameters. While the core architecture remained the same across datasets (MNIST, Medical MNIST, CIFAR-10), adjustments were made for input size, number of convolutional layers (2 for Medical MNIST, 3 for CIFAR-10), and output neurons to match class counts. The number of **qubits and quantum layers remained constant** in all tests.

RESULT

- **HQNN-Parallel achieved 99.21% accuracy.**
- **Classical CNN (with the same convolutional base) achieved slightly lower accuracy and required 8× more parameters.**
- **No transfer learning or pretrained models were used.**
- **Same architecture as MNIST, with minor adjustments to input size and output layer.**
- **Achieved >99% accuracy, confirming generalizability to medical image data.**
- **HQNN-Parallel achieved 82.78% test accuracy, slightly outperforming the classical CNN at 82.64%.**
- **Despite similar accuracy, the HQNN used fewer trainable parameters.**
- **Adjustments included 3 convolutional layers and modified output neurons for 10 classes.**
- **Number of qubits and quantum layers remained constant.**

2.Applications of Hybrid Quantum Models

Hybrid quantum models, including HQNNs, have shown improvements in image classification tasks on datasets like MNIST, Medical MNIST, and CIFAR-10. Quantum layers enable better feature extraction, leading to higher accuracy with fewer parameters compared to classical models.

3.Performance Comparison of HQNNs with CNNs

Studies have shown that HQNNs outperform classical CNNs in terms of accuracy and parameter efficiency. For example, the HQNN-Parallel model achieved 99.21% accuracy on

MNIST, surpassing a classical CNN with 8x more parameters. It also showed strong generalizability on Medical MNIST and CIFAR-10.

4.Challenges and Future Directions

Quantum models face challenges such as noise, decoherence, and limited qubit availability. However, advancements in quantum hardware and optimization algorithms are addressing these limitations, and future research will focus on improving the efficiency and scalability of quantum models for real-world applications.

5.Conclusion

Hybrid quantum-classical models, especially HQNNs with quantum convolutional layers, offer significant advancements in image classification. These models achieve high accuracy with fewer parameters, and as quantum computing technology evolves, they will likely become more practical for solving complex image classification challenges.

Literature Review

1. Quantum Computing Foundations

Quantum computing is a paradigm shift from classical computing, built on principles of quantum mechanics such as superposition, entanglement, and quantum interference. Unlike classical bits that exist in states 0 or 1, quantum bits (qubits) can exist in linear combinations of both states simultaneously. This property enables exponential growth in computational space as the number of qubits increases.

Early theoretical models of quantum computation were introduced by Richard Feynman (1982) and David Deutsch (1985), proposing quantum systems as more efficient alternatives for simulating physical systems. Since then, various quantum algorithms like Shor's algorithm for factoring and Grover's algorithm for searching have demonstrated theoretical speedups over their classical counterparts.

2. Image Processing and Classification Techniques

Image processing involves a sequence of operations for noise removal, edge detection, feature extraction, and object recognition. With the rise of deep learning, Convolutional Neural Networks (CNNs) have become the de facto standard for image classification tasks. CNNs use convolutional layers to extract spatial hierarchies of features followed by dense layers for classification.

Despite their success, CNNs face issues such as:

- Overfitting due to high model complexity.
- High computational cost for large datasets.
- Limited generalization in small-sample or noisy scenarios.

These limitations motivate the exploration of quantum-enhanced models, which potentially provide:

- Exponential feature space growth for complex pattern representation.
- Better generalization from fewer parameters.
- Improved optimization through non-classical gradient landscapes.

3. Quantum Approaches to Image Classification

Quantum-enhanced models are particularly well-suited for image classification due to their ability to represent and manipulate data in high-dimensional Hilbert spaces. Quantum circuits can encode pixel intensity values into quantum states, where entangled gates capture global patterns across the image.

Studies such as Mari et al. (2020) and Havlíček et al. (2019) demonstrated how VQCs can be trained to classify image data with comparable or better accuracy than classical models under certain conditions. Research also shows that using quantum feature maps in combination with classical layers can boost performance while reducing the number of tunable parameters.

Furthermore, quantum data re-uploading techniques allow classical inputs to be encoded multiple times across quantum layers, increasing model expressiveness without requiring deeper circuits.

4. Quantum Machine Learning (QML)

4.Quantum Machine Learning (QML)

Quantum Machine Learning (QML) is an interdisciplinary field combining quantum computing and machine learning. The primary goal is to leverage quantum properties to enhance data representation, model expressivity, and computational speed. There are three main approaches in QML:

Quantum Data and Classical Processing (QDAC) – where quantum-generated data is used in classical models.

Classical Data and Quantum Processing (CDQP) – classical data is encoded into quantum states and processed via quantum circuits.

Hybrid Models – combining classical neural network components with quantum circuits, known as Hybrid Quantum Neural Networks (HQNNs).

Significant contributions in QML include the development of Variational Quantum Circuits (VQCs) and Quantum Support Vector Machines (QSVMs). Researchers like Schuld et al. have shown how quantum-enhanced models can be trained using classical optimizers to reduce quantum circuit depth, improving feasibility on noisy intermediate-scale quantum (NISQ) devices.

5. Hybrid Quantum Neural Networks (HQNN)

5.Hybrid Quantum Neural Networks (HQNN)

HQNNs represent a promising architecture that leverages classical deep learning techniques alongside quantum computation. The classical part typically handles tasks such as feature extraction (e.g., using convolutional layers), while quantum layers perform tasks that benefit from high-dimensional representation, such as classification or decision-making.

Studies by Tacchino et al. (2020) and Farhi et al. (2018) introduced architectures where classical outputs are encoded into quantum circuits using angle encoding or amplitude encoding, processed through entangled quantum gates, and then measured for final predictions. HQNNs are particularly suited for image classification, where the dimensionality and patterns within image data align well with the complex feature spaces offered by quantum processing.

Recent work has shown promising results using platforms like Pennylane, Qiskit, and TensorFlow Quantum, enabling simulation and training of hybrid models using standard datasets such as MNIST and Fashion-MNIST.

6. 5.Challenges and Future Directions

Despite promising results, several challenges remain:

Quantum hardware limitations: Current NISQ devices are prone to noise and decoherence, which can impact reliability.

Data encoding: Efficiently mapping classical data (e.g., pixel intensities) into quantum states remains an open problem.

Scalability: As image resolution increases, encoding more features into limited qubit registers becomes difficult.

Benchmarking: There is a need for standardized benchmarks to compare HQNNs against classical and purely quantum models.

Future work focuses on:

Developing deeper and more expressive quantum circuits.

Integrating quantum convolutional layers.

Deploying HQNNs on real quantum hardware.

Extending hybrid models to multi-class classification and real-world image datasets (e.g., CIFAR-10, medical imaging).

References

1. https://www.researchgate.net/profile/Bahman-Zohuri/publication/356468274_What_is_Quantum_Computing_and_How_it_Works/links/619fb463c4ed925f9debd181/What-is-Quantum-Computing-and-How-it-Works.pdf
2. https://www.ibm.com/think/topics/quantum-computing
3. https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-quantum-computing#Real-world-uses
4. https://www.linkedin.com/pulse/brief-history-quantum-computing-luke-antal
5. www.cl.cam.ac.uk
6. https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-quantum-computing/
7. https://thequantuminsider.com/2023/03/24/quantum-computing-challenges/
8. https://chatgpt.com/
9. https://medium.com/@roysuman088/quantum-image-processing-the-future-of-visual-data-manipulation-3b646bb0ccfa
10. https://www.google.com/url?sa=i&url=https%3A%2F%2Fphyslab.org%2Fwp-content%2Fuploads%2F2023%2F04%2FQuantumImageRepresentation_22120005.pdf&psig=AOvVaw06Z_cyVFn6WFmp4lUkokN0&ust=1746629448519000&source=images&cd=vfe&opi=89978449&ved=0CAQQn5wMahcKEwi41YifjI-NAxUAAAAAHQAAAAAQBA
11. https://link.springer.com/article/10.1007/s11831-021-09599-2
12. https://www.youtube.com/watch?v=tJMe2kiNWR0
13. https://arxiv.org/abs/2203.01831
14. https://arxiv.org/abs/2304.09224
15. https://www.ibm.com/think/topics/quantum-computing
16. https://pennylane.ai/qml/demos/tutorial_sc_qubits
17. https://www.nisenet.org/content-keywords/quantum-dots
18. https://iopscience.iop.org/article/10.1088/2632-2153/ad2aef/meta