Transactions on Engineering and Computer Science

Research| Vol 3 Iss 1

# A Post-Quantum Blockchain Application in M-band Wavelet and Fresnel Domain: A Steganography-Based, Decentralized, Distributed Ledger System

**Sonok Mahapatra[1*], Tyler Wooldridge[2], and Xiaodi Wang[2]**

[1]Danbury Math Academy Danbury, CT, USA

[2]Western Connecticut State University, Danbury CT, USA

*Corresponding author: Mahapatra S, Danbury Math Academy Danbury, USA; E-mail: sonokmahapatra[AT]gmail.com

Received: December 25, 2021; Accepted: February 23, 2022; Published: March 24, 2022



## Abstract

***Following the information age, a new computational data structure known as blockchain has risen to prominence as an open, public, distributed ledger with a wide range of uses. With applications in secure sharing of medical data, voting mechanisms, cross-border payments, personal identity security, and most notably cryptocurrency exchange, blockchains seek to revolutionize how we handle our data fundamentally. On the other hand, the technological development of quantum computers has opened several vulnerabilities to numerous blockchain applications. Therefore, improper methods of establishing privacy for blockchain can compromise large amounts of user data, making the development of high-level privacy-preserving mechanisms impervious to quantum computing of great importance. In this research, we developed three security schemes for quantum computing-resistant blockchain applications. The first security scheme was the use of classical-quantum mappings for zero-knowledge proof of data. The second security scheme proposed was an optical encryption scheme for information security under the basic double random phase encoding framework using enhanced complexity and immunity in the Fresnel domain. The final mechanism we proposed is a wavelet-based steganography scheme for increased storage, concealment, and limited accessibility. We used a wavelet domain to transmit pseudo-quantum signals in RGB color QRs for robust and secure data encryption capabilities.***

*Keywords:* ***Blockchain; m-band wavelet transfor; Steganography; Pseudo quantum signaling; Quantum signaling; Grover's algorithm; Shor's algorithm; Asymmetric key cryptography; Fresnel domain; Fresnel diffraction; Quantum algorithm; Reverse encryption; Color QRs; Quantum cryptography; Transaction verification; Proof of work; Post-quantum cryptography; and Optical encryption scheme.***

## 1. Introduction

More in today's world than ever, the establishment of a stable and efficient network for recording data is paramount, for in the wake of the information age, significant applications are becoming more and more digitized, and the threat of information leakage threatens significant aspects of a society which rely on information confidentiality [38], [47].

**Citation:** Mahapatra S, Wooldridge T, and Wang X. A Post-quantum blockchain application in m-band wavelet and Fresnel domain: A steganography-based, decentralized, distributed ledger system. Trans Eng Comput Sci. 2022;3(1):126.

The mishandling of a data pool can jeopardize millions of people's data, including IP addresses, names, date of birth, address, social securities, banking information, medical data, housing information, transactions, records, and so on. Examples of this include the Anthem Blue Cross breach that put about 80 million individuals' medical data at risk or Yahoo's data breach subject to the most significant data breach in human history in which a record 3 billion of its user accounts were impacted.

As a result of this increased reliance on computer systems, the area of cybersecurity is becoming increasingly important and internet devices often for data management due to the growth in the number of intelligent devices, the integration of computer sciences and data retrieval, and the growth of numerous applications that use the internet and rely on information technology. Unfortunately, due to the increasing threat to establishing robust and efficient security methods for privatized data, researchers, mathematicians, government agencies, and computer scientists among others are left to find a solution [1].

One significant technological development, as of late, is the rapid development of quantum computers [6], [8]. Quantum computers and quantum algorithms employ quantum mechanics that have the capability to solve intractable problems, that secure most internet commerce, and data stores today [17]. Consequently, finding secure data management, storage, and transmission algorithms that are impervious to quantum computers is of monumental importance.

This paper uniquely employs a wide array of classical, quantum, and theoretical models to approach the development of a post-quantum blockchain network, with extensive mathematical and programmatic research to support these developments [9], [31]. Furthermore, the paper clearly illustrates the advancement of previous ideas in cybersecurity, mathematics, physics, quantum mechanics, among other areas to establish a robust and efficient framework.

### 1.1 Prior works

The term post-quantum blockchain is defined as a set of security algorithms to implement a blockchain data structure that is thought to be resistant against a cryptanalytic attack by a quantum computer. However, as of 2021, most blockchain applications can be efficiently compromised by a sufficiently powerful quantum computer. Over the last decade, researchers have been developing different mechanisms to establish privacy-preserving methods to improve characteristics, such as implementing public-key cryptography and hash functions [1]. The improvements of such privacy mechanisms fall under the study of post-quantum cryptography, which refers to cryptographic algorithms (usually public-key algorithms) that are thought to be secure against a cryptanalytic attack by a quantum computer [8], [9].

Quantum key distribution is a well-known privacy-establishing post-quantum cryptographic scheme. QKD is a cryptographic protocol that allows two parties to generate a shared random secret key using quantum biases known only to them via quantum entanglement. These biases can then be used to encrypt and decrypt messages. Another post-quantum algorithm is Optimal Asymmetric Encryption Padding (OAEP) [49] which is a kind of Feistel network that employs a pair of random oracles G and H to process plaintext data prior to asymmetric encryption that's then put through a one-way trapdoor permutation *f*. As a consequence, a combination technique that is secure against the

plaintext attacks is created. Multivariate cryptography [50], a post-quantum cryptographic approach based on multivariate polynomials over a finite field F, is another cryptographic system. The key security assumption is that solving nonlinear multivariate equations over a finite field is NP-hard. [2], [5], [24], [31], [40], contains a list of other methods.

## 2. Background

### 2.1 Blockchain

#### 2.1.1 Key technologies

**Hash algorithm:** A hash algorithm is a function that converts data into a numeric string output of fixed length called a hash value. This hash value is often used to verify data integrity, such as in the proof of work algorithm. These hash values are designed to be collision-resistant, making the probability of different data creating the same hash value doubtful and difficult to find [1]. The hash value of data is stored in the block of a blockchain. In addition, signatures to verify transactions are commonly generated from the hash of an owner's private key and the data that needs to be signed [6], [51].

**Proof of work:** Proof of work (POW) is a cryptographic-based zero-knowledge proof system that both proves some data exists and that there is some computational effort done to support the authenticity of the data. In a blockchain system, any node that wants to generate a new block must write all transactions to the blockchain and do a proof of work puzzle to generate the block. Users are incentivized to mine and verify transactions in the blockchain for a reward. The network relies on the blockchain, with the most work done to verify and transmit block information to other nodes in the network [16], [21] (Fig. 1).

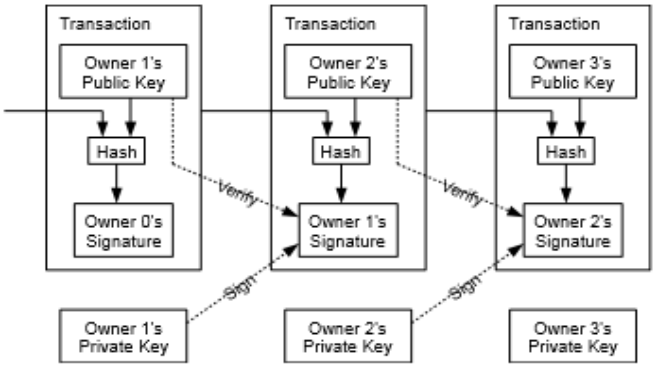


**Fig. 1.** Proof of work [54].

**Timestamp:** Timestamps are used to address the issue of double-spending. The blockchain system publishes the hash block of items combined with a timestamp server and utilizes the timestamp network to establish that the data must have existed at the moment, which implies that the ownership of the money in a transaction cannot be transferred again. Each hash contains the preceding timestamp, establishing a chain, with each block strengthening the ones before it [1] (Fig. 2).

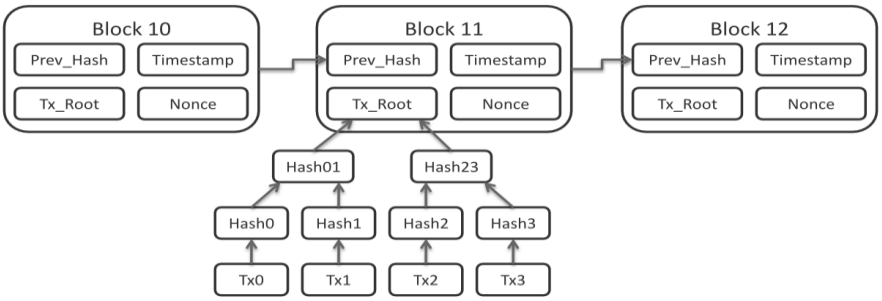


**Fig. 2.** Blockchains [55].

### 2.1.2 Design of blockchain

**Structure of blockchain:** The blockchain system is made up of a linked list of data blocks that keep track of all data transactions in sequential order. Each block carries a timestamp, a hash value for its data, and the hash value of the preceding block. Blocks are generated in chronological order, and once it is confirmed to be valid, it is challenging to modify [1], [37].

**Network of blockchain:** Blockchain network: In Nakamoto, the stages to running a network such as a blockchain are as follows:

- All nodes in the network are notified of new transactions.
- Each node collects transactions into a block.
- When a user on a network node solves a proof of work challenge, the block is broadcast to all nodes.
- Other nodes will only accept the block if the transactions are legitimate and have not been spent before.
- The next block in the chain is generated using the hash of the accepted block as the preceding hash, demonstrating that all nodes in the network accept the block (Fig. 3).

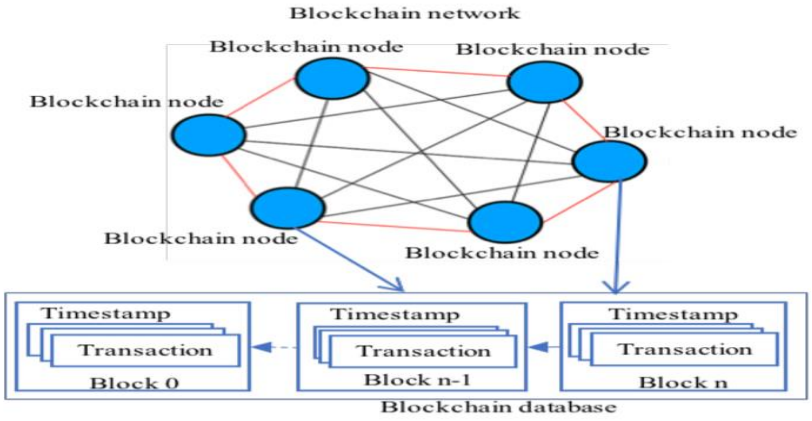


**Fig. 3.** Blockchain network [56].

## 2.2 Grover's Algorithm

**Introduction and importance:** Grover's algorithm is a quantum computing search algorithm for an unstructured search that finds with high probability the unique input given the particular output and BlackBox function. Grover's

algorithm uses just $O(\sqrt{N})$ evaluations, whereas a classical computer requires $O(N)$ passes, where *N* is the size of the function's domain [6], [10]. Even if Grover's method does not give an exponential speedup over conventional computers when compared to other quantum computing techniques (such as Shor's algorithm), it might be utilized to speed up a wide variety of algorithms [32], [33]. Particularly for NP-complete problems which contain extensive search, Grover's algorithm can speed up. Applications of Grover's algorithm include provable speedups for black-box problems in a quantum query, including the collision problem. Consequently, because of the function inversion ability of Grover's algorithm, the threat of collision attacks and preimage attacks which are fundamental to asymmetric cryptography and blockchain, is heightened dramatically [10], [31].

**Design**

The steps of Grover's algorithm are the following

Given a function *f* and oracle $U_f$ in the form of a unitary operator such that

$$U_f|x\rangle \ |y\rangle \ = |x\rangle \ |y \oplus f(x)\rangle \ , \qquad (2.2.1)$$

We may implement $U_w$ when $U_f$ has its control qubit in a state of equal super-positions (denoted as $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle = H|1\rangle)$. This can be written as

$$U_f(|x\rangle \oplus \ |-\rangle) = (U_w|x\rangle) \oplus |-\rangle \quad (2.2.2)$$

We also define Grover diffusion operator to be

$$U_s = 2|s\rangle\langle s| - I \quad (2.2.3)$$

Finally, we define $r(N) \leq \frac{\pi}{4}\sqrt{N}$ , to be the number of times we wish to implement the amplification circuit.

1) We put all qubits into uniform superposition using a Hadamard gate.

$$|S\rangle = \frac{1}{\sqrt{N}} \Sigma \sum_{i=0}^{N-1} |x\rangle \qquad (2.2.4)$$

1) We mark the state we wish to find by performing a phase flip that inverts a state's amplitude which can be done by implementing an amplification circuit that further increases the marked state's amplitude while decreasing the amplitude of all other states.

This is done by first applying the operator $U_w$ , then Grover's diffusion operator , $U_s$, $r(N)$ times (Fig. 4).

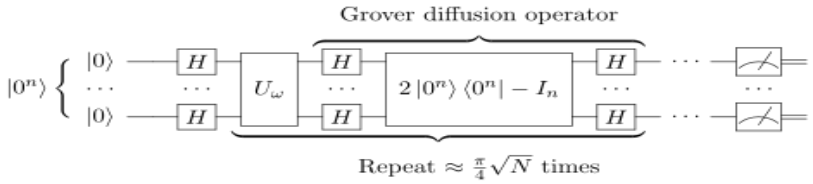


**Fig. 4.** Subroutine utilized in Grover's algorithm [57].

## 2.3 Shor's Algorithm

**Introduction and importance:** Shor's algorithm is a quantum computing algorithm designed for integer factorization that runs in polynomial time. On a quantum computer to factor [6], an integer, N, Shor's algorithm runs in order of $O((logN)^2\,(loglogN)(logloglogN))$, whereas a classical computing algorithm, say the general number field sieve , would run at best in subexponential time with order of $O(e^{1.9(logN)1/3(loglogN)2/3})$. Shor's algorithm is derived from the quantum Fourier transform and modular exponentiation. Shor's algorithm is important because it can decrypt RSA encryption [15], [49] more efficiently, which is often fundamental to signature technology and is crucial to the authenticity of block transactions and, therefore, the network [8].

**Design:** The steps of Shor's algorithm [11] are the following:

We define Shor's algorithm for some number N such that function $f(x) = a^x \bmod N$ and $Q = 2^q$, For $N^2 \le Q < 2N^2$. We also define r to be the finite order of $a \bmod N$ such that r is the smallest positive integer in which $a^r \equiv 1 \bmod N$. In order for the input and output qubits to hold a superposition of values from 0 to Q-1, we must have q qubits.

First, we initialize the registers, which are at an initial state of $|0\rangle$ , to states of superposition as shown:

$$\frac{1}{\sqrt{Q}}\sum_{x=0}^{Q-1}|x\rangle \;=\; \left(\frac{1}{\sqrt{2}}\sum_{x_1=0}^{1}|x_1\rangle\right)\oplus\left(\frac{1}{\sqrt{2}}\sum_{x_1=0}^{1}|x_1\rangle\right)\oplus\ldots\oplus\left(\frac{1}{\sqrt{2}}\sum_{x_q=0}^{1}|x_q\rangle\right) \tag{2.3.1}$$

Next, we construct an oracle to obtain, $U_f\,|x,0^q\rangle \;=\; |x, f(x)\rangle$, which gives us the following:

$$U_f\frac{1}{\sqrt{Q}}\sum_{x=0}^{Q-1}|x,0^q\rangle \;=\frac{1}{\sqrt{Q}}\sum_{x=0}^{Q-1}|x,f(x)\rangle \tag{2.3.2}$$

Using twice as many qubits as necessary guarantees at least N different values that produce the same f(x). Next, we apply the inverse quantum Fourier transform to the input register in which we use a Qth root of unity in the form of $\omega = e^{\frac{2k\pi i}{Q}}$ to distribute the amplitude of a given quantum state among all Q of the $|y\rangle$ states. We, therefore, obtain the following:

$$U_{QFT}(|x\rangle\ ) = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} \omega^{xy} |y\rangle \tag{2.3.3}$$

This leaves us with the following superposition of more than Q states, but fewer than $Q^2$ states

$$\frac{1}{Q} \sum_{x=0}^{Q-1} \sum_{x=0}^{Q-1} \left[ \sum_{x\ \varepsilon\ \{0,\ \dots\ ,\ Q-1\};\ f(x)=z} \omega^{xy} \right] |y, z\rangle \tag{2.3.4}$$

Let $x_0$ be the smallest of the $x\ \varepsilon\ \{0, \dots, Q-1\}$. Then we can write m such that $m - 1 = \lfloor \frac{Q-x_0-1}{r} \rfloor$, and b such that $x_0 + rb < Q$. We obtain the final state of the coefficient of $\frac{1}{Q}|y, z\rangle$ :

$$\sum_{x\ \in\ \{0,\ \dots\ ,\ Q-1\};\ f(x)=z} \omega^{xy} = \sum_{b=0}^{m-1} \omega^{(x_0+rb)y} = \omega^{x_0 y} \sum_{b=0}^{m-1} \omega^{rby} \tag{2.3.5}$$

Each term in the sum represents a different path to the same result which means that quantum interference now may occur. We may then measure our outcome. Since f is periodic, the probability of measuring some state $|y, z\rangle$ is given by:

$$Pr(|y, z\rangle\ ) = \left| \frac{1}{Q} \sum_{x\ \varepsilon\ \{0,\dots,Q-1\};\ f(x)=z} \omega^{xy} \right|^2 = \frac{1}{Q^2} \left| \sum_{b=0}^{m-1} \omega^{(x_0+rb)y} \right|^2 =$$

$$\frac{1}{Q^2} |\textstyle\sum_{b=0}^{m-1} \omega^{rby}|^2 = \frac{1}{Q^2} \left| \frac{\omega^{mry}-1}{\omega^{ry}-1} \right|^2 = \frac{1}{Q^2} \frac{sin^2\left(\frac{\pi mry}{Q}\right)}{sin^2\left(\frac{\pi ry}{Q}\right)} \blacksquare \tag{2.3.6}$$

We can then observe that the probability converges to the unit vector $\omega^{ry}$ that is closer to the positive real axis or $\frac{yr}{Q}$ is to an integer, from which we see that s is very likely to be the appropriate order r or at least a factor of it. If $f(x) = f(x + s)$ we're done; otherwise we try for other multiples of s or try again from step 1 (Fig. 5).

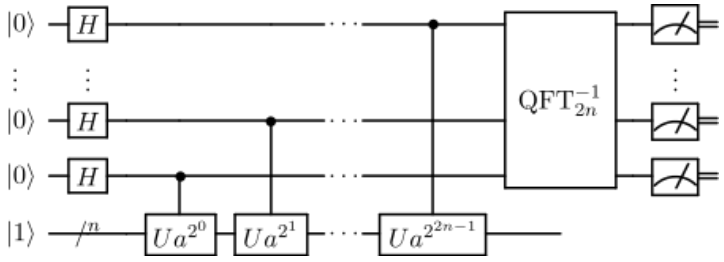


**Fig. 5.** Subroutine utilized in Shor's algorithm [58].

## 2.4 Discrete M-band Wavelet Transform (DMWT)

Discrete M-band wavelet transforms (DMWT) use M filter banks (one of which is a lower filter-bank while the rest are high pass filter-banks) where M ≥ 2, to decompose a K-dimensional signal into M different frequency levels. [3][5] In

this research, we use a 4-band wavelet transform to transform input data X, into 4 frequency levels including a low-frequency/approximation component and 3 high-frequency/detail components.

In Multiresolution Analysis, the theory about wavelets, a low-pass filter bank forms linearly independent vectors that span the approximation space $V_i$, while the high-pass filter banks form detail spaces $W_i$, $i = 1 \ldots M-1$. Each approximation space can be decomposed into higher approximation and detail sub-spaces

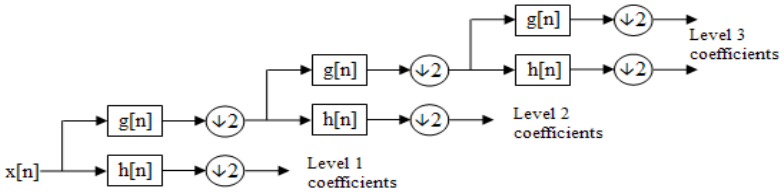


**Fig. 6.** Diagram of the wavelet transform [59].

At each level, the wavelet transform [5] creates the approximation space and detail space.
For instance, a 4 band wavelet approximation space can be decomposed as

$$\mathbb{R}^N = V_0 = V_1 \oplus W_{1,1} \oplus W_{1,2} \oplus W_{1,3}. \quad (2.4.1)$$

We can further decompose this as

$$\mathbb{R}^N = V_3 \oplus W_{3,1} \oplus W_{3,2} \oplus W_{3,3} \oplus W_{2,1} \oplus W_{2,2} \oplus W_{2,3} \oplus W_{1,1} \oplus W_{1,2} \oplus W_{1,3} \quad (2.4.2)$$

Let an M-band wavelet have filter banks $\alpha^{(1)}$, $\beta^{(1)}$, ..., $\beta^{(M-1)}$. Then the filter banks have the following properties:
For $m = 1, \ldots, M-1$ :

$$1: \|\alpha_i\| = \|\beta^{(1)}\| = \|\beta^{(2)}\| = \ldots = \|\beta^{(M-1)}\| = 1 \quad (2.4.3)$$

$$2: \sum_{i=1}^{N} \alpha_i = \sqrt{M}, \sum_{i=1}^{N} \beta_i^{(m)} = 0 \quad (2.4.4)$$

$$3: \alpha \cdot \beta^{(m)} = 0 \quad (2.4.5)$$

Where N is the length of the filter bank, additionally if the M-band wavelet is k-regular, for
$j = 0, \ldots, K-1$.

$$4: \sum_{i=1}^{N} i^j \cdot \beta_i^{(m)} = 0 \quad (2.4.6)$$

$$5: T^t = T^{-1}, \quad (2.4.7)$$

where T is the wavelet transform matrix, and

$$6: \beta^{(i)} \cdot \beta^{(j)} = 0, \text{for all } 1 \le i < j < M-1. \quad (2.4.8)$$

An example of 4-band wavelet transform matrix T is given below [14]:

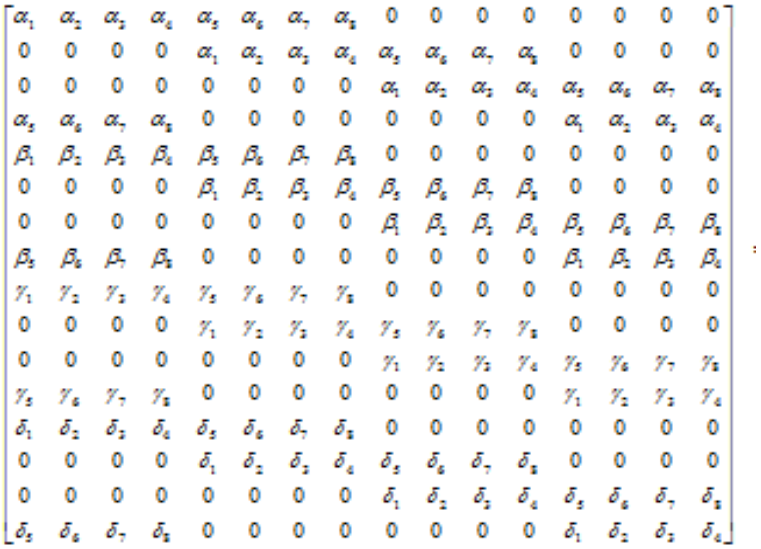

**Fig. 7.** 4-Band wavelet transform [60].

For our two functions, we use a 4 band orthonormal wavelet to decompose our dataset into 4 frequency levels. We use the 4 filter banks as shown in [3], [22], Table 1 below.

**Table 1:** Filter Banks for the 4-Band Wavelet Transform.

| $\alpha$ | $\beta^{(1)}$ | $\beta^{(2)}$ | $\beta^{(3)}$ |
|---|---|---|---|
| $\alpha_1$ =-0.0674 | $\beta_1^{(1)}$ =-0.0942 | $\beta_1^{(2)}$ =-0.0942 | $\beta_1^{(3)}$ =-0.0674 |
| $\alpha_2$ =0.0942 | $\beta_2^{(1)}$ =0.0674 | $\beta_2^{(2)}$ =-0.0674 | $\beta_2^{(3)}$ =-0.0942 |
| $\alpha_3$ =0.4058 | $\beta_3^{(1)}$=0.5674 | $\beta_3^{(2)}$ =0.5674 | $\beta_3^{(3)}$ =0.4058 |
| $\alpha_4$ =0.5674 | $\beta_4^{(1)}$=0.4058 | $\beta_4^{(2)}$ =-0.4058 | $\beta_4^{(3)}$ =-0.5674 |
| $\alpha_5$ =0.5674 | $\beta_5^{(1)}$=-0.4058 | $\beta_5^{(2)}$ =-0.4058 | $\beta_5^{(3)}$ =0.5674 |
| $\alpha_6$ =0.4058 | $\beta_6^{(1)}$=-0.5674 | $\beta_6^{(2)}$ =0.5674 | $\beta_6^{(3)}$ =-0.4058 |
| $\alpha_7$ =0.0942 | $\beta_7^{(1)}$=-0.0674 | $\beta_7^{(2)}$ =-0.0674 | $\beta_7^{(3)}$ =0.0942 |
| $\alpha_8$ =-0.0674 | $\beta_8^{(1)}$=0.0942 | $\beta_8^{(2)}$ =-0.0942 | $\beta_8^{(3)}$ =0.0674 |

It's easy to verify

$$\sum_{i=1}^{8} \alpha_I = \sqrt{4} = 2, \sum_{i=1}^{8} \beta_i^{(m)} = 0, \|\alpha_i\| = \|\beta^{(1)}\| = \|\beta^{(2)}\| = \dots = \|\beta^{(M-1)}\| = 1, \quad (2.4.9)$$

$$\alpha \cdot \beta^{(1)} = \alpha \cdot \beta^{(2)} = \alpha \cdot \beta^{(3)} = \beta^{(1)} \cdot \beta^{(2)} = \beta^{(1)} \cdot \beta^{(3)} = \beta^{(2)} \cdot \beta^{(3)} = 0 \quad (2.4.10)$$

## 2.5 Fresnel Diffraction

In optics, the Fresnel diffraction equations are for the near field diffraction phenomenon, which can be applied to the propagation of waves in near field electromagnetic fields around an object. Fresnel diffraction occurs when an electron source or some observation point is placed at a limited distance from an item, and the result is viewed as spherical waves dispersed from the edge of a specimen that interferes with the incident wave [14] ( they are observed as an interference fringe or Fresnel fringe). In this research, we use a Fresnel transform (the diffraction of waves in the Fresnel wave domain) to reconstruct an image from a phase-only mask with the various parameters, including Fresnel wavelength transforms and propagation distances between an object and an image plane [25]. We use the Fresnel transform as it has numerous valuable properties, including duality, translation, dilation, and unitary [28]. This unitary property provides a perfect reconstruction of an image; the duality property allows us to compute the inverse Fresnel transform and gives the ability to denote the amplitude and phase truncated constructions of waves in the Fresnel domain [27] (Fig. 8).

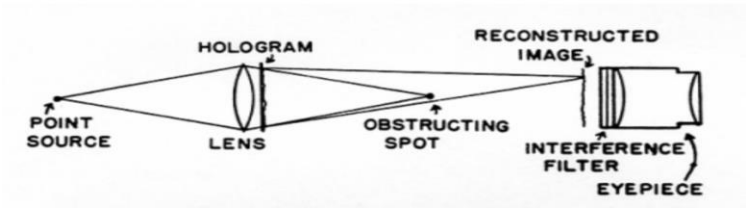


**Fig. 8.** Fresnel diffraction [61].

The Fresnel diffraction [27] (an approximation of Fresnel-Kirchhoff diffraction that can be applied to propagate waves in the near field) can be simplified as the following:

$$E(x_0, y_0) \propto \int\int exp\left\{jk\left[\frac{(-2x_0x_1 - 2y_0y_1}{2z} + \frac{(x_1^2 + y_1^2)}{2z}\right]\right\} Aperture(x_1, y_1)dx_1dy_1, \quad (2.5.1)$$

So that the plane of the aperture is represented by $x_1, y_1$, while the plane of observation is represented as $x_0, y_0$, and z as the length downstream. [36]

## 2.6 Color QR Codes

A QR code is a type of matrix barcode that can be interpreted and extracted for some pattern present in both the horizontal and vertical components of the image. In this research, we use a stack red, blue, green (RGB) schema intending to develop covert, colored QR codes in which the capacity to store information is increased, and security is enhanced (Fig. 9).

**Fig. 9.** Example of a color QR code [62].

In this research, we represent every pixel as an array [22], [23] with each pixel array being represented as [R, G, B], defining some intensity of red, some intensity of green, and some intensity in blue (with 256 various intensities per color channel though more colors and levels could be used). Since all colors can be spanned by the orthonormal color basis of

$$Red([1,0,0]), Green([0,1,0]), Blue([0,0,1]) \; \varepsilon \; Span \; R^3, \qquad (2.6.1)$$

Each color QR code can be stacked and split into these three color channels [41] (Fig. 10).

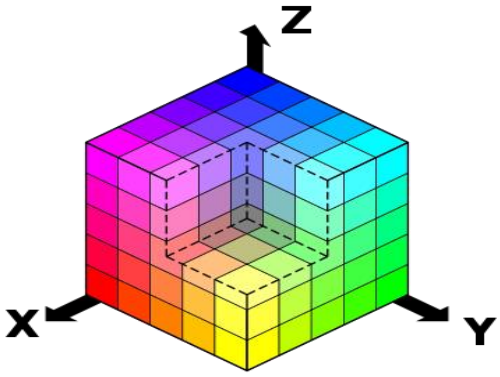


**Fig. 10.** Separation into color channels [63].

## 2.7 Quantum Computing

A quantum computer is a device used for quantum computation that makes direct use of quantum mechanics, which is believed to solve certain computational problems. So, while normal computers store information in classical bits that exist in a state of 0 or 1, quantum computers use quantum bits or qubits, which exist in a superposition of both states. [2][3] Meaning that a qubit can be both $|0\rangle = [0,1]^T$ or $|1\rangle = [1,0]^T$ as well as any linear combination of them. Qubits can be expressed as $|\psi\rangle = a|0\rangle + b|1\rangle$, where a and b are complex numbers in which $|a|^2 + |b|^2 = 1$. Qubits may also be expressed as a point on the surface Bloch sphere, in which there are infinite states for a qubit as shown (Fig. 11):

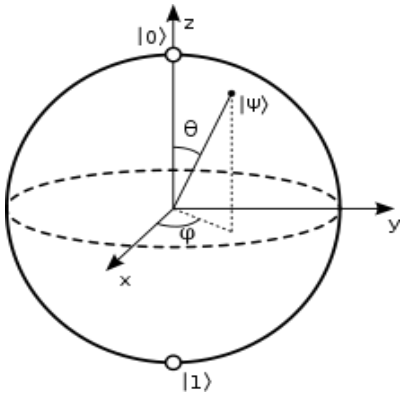


**Fig. 11.** The bloch sphere [64].

Geometrically we can express the state of a qubit as, $|\psi\rangle = cos\frac{\theta}{2}|0\rangle + e^{i\varphi}cos\frac{\theta}{2}|1\rangle$ , where $\theta$ and $\varphi$ are real numbers. When we measure a quantum state because of quantum decoherence, when we measure a qubit in superposition, it collapses into a probability of 0 or 1. When measured in accordance with Schrodinger's equation, $ih\frac{\partial}{\partial t}|\psi\rangle = \hat{H}|\psi\rangle$ , [42], [52] we get that the probability of the outcome $|0\rangle$ with value 0 being $|\alpha|^2$ and the probability of outcome $|1\rangle$ with value 1 being $|\beta|^2$.

## 2.8 Pseudo Quantum Signals

In this research, we propose a method to transform signals into a set of pseudo-quantum signals [2], [3], [20]. For some signal $S = \{s_i\}^N_{i=1}$, we define a linear pseudo quantum signal converter as:

$$F(s^*) = \frac{m\pi}{3} \text{and } F(s_*) = \frac{n\pi}{6}, \text{where } s^* = max(s_i), s_* = min(s_i), m \in N \text{ and } n \in N. \quad (2.8.1)$$

We convert the signal S into an interval $[\frac{n\pi}{6}, \frac{m\pi}{3}]$, with $\theta_i = F(s_i)$ for all i = 1, 2, 3, …, N. We can further express our transformed result, F(s), as a signal value pseudo qubits

$$|s_i\rangle = cos(\theta_i)|0\rangle + sin(\theta_i)|1\rangle. \quad (2.8.2)$$

Then by definition, this pseudo-quantum signal can be processed by a classical computer to simulate a quantum space. Hence, these signals are essential when only a classical computer is used to simulate quantum signals and quantum mechanisms.

## 2.9 Asymmetric Key Cryptography

Public-key cryptography, also known as asymmetric cryptography, is a cryptographic system that uses a pair of keys, a public key known to others, and a private key known only by the owner. In the system, any person can encrypt a message using the intended receiver's public key that can only be decrypted using the receiver's private key. [21] The scheme has the advantage of not manually pre-sharing symmetric private keys, making robust authentication possible (Fig. 12).

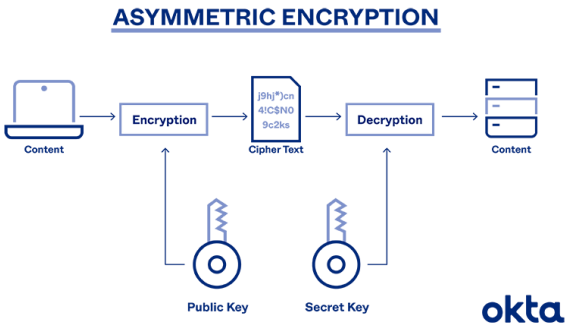


**Fig. 12.** Asymmetric encryption protocol [65].

The generation of such key pairs depends on the generation of one-way functions (for instance, the ease of multiplication of large numbers and difficulty of factorizations of such large numbers used in the RSA encryption, [16], [30] which is used in most encryption applications).

Public key encryption algorithms are fundamental to modern cryptographic systems, including applications that ensure confidentiality and authentication of electronic communications and data storage [6], such as the signature technology used in numerous blockchain applications.

## 3. Privacy-Preserving Mechanisms

### 3.1 Quantum Mechanism for Zero-Knowledge Proof

#### 3.1.1 Permissions

The current cryptographic hash functions widely used by blockchain applications are used to pass information anonymously between users. These hash functions act as privacy-preserving mechanisms that ensure data anonymity and usage in data storage and retrieval. However, hash algorithms suffer from Grover's Algorithm [6], [8], [10] because they produce a fixed-size output given any random-sized input. The augmented speed of Grover's algorithm can be used to expedite the collision attack and, therefore, compromise the blockchain's security [18].

In order to create a zero-knowledge-proof system that is resistant to quantum computer algorithms, we turn to quantum information theory [40], [42], [44]. In quantum information theory, we have a quantum channel that acts as a communication mechanism that can transmit quantum information that can be manipulated using quantum information processing techniques [41], [43]. Due to the following quantum properties such as the no-teleportation theorem (which states that qubits cannot be observed in a classical state), no-cloning theorem (which prevents a quantum bit in superposition from being copied), no-deleting theorem [43] (which prevents a quantum bit from being deleted), no-broadcast theorem (which prevents an arbitrary qubit from being delivered to multiple recipients), and the no-hiding theorem (where quantum bits retain their information) quantum information theory has broad applications of quantum communications.

### 3.1.2 Steps to load classical data into a quantum channel

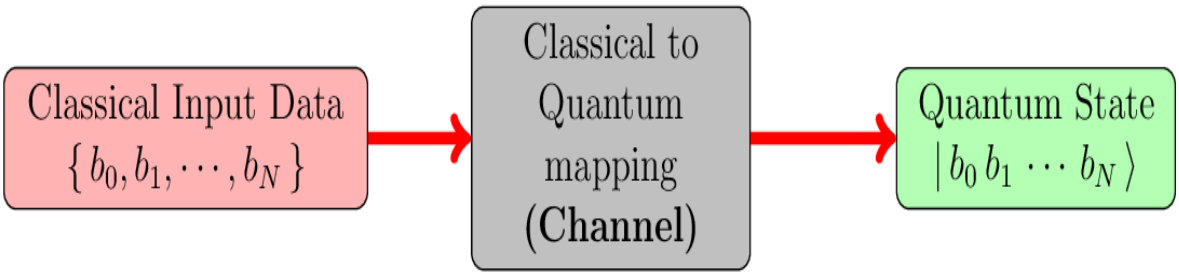


**Fig. 13.** Process to create a quantum channel [66].

To create a quantum channel we must first format our classical data into a quantum computing input format [44]. We would then have to create a theoretical classical to quantum mapping that outputs a quantum state of entangled qubits (which can be expressed as $|b_o b_1 \ldots b_n\rangle = |b_o\rangle \otimes |b_1\rangle \otimes \ldots \otimes |b_n\rangle$ ). We can further improve on our quantum channel by packing N classical bits into a quantum state of size $log_2(N)$ qubits, which is enabled by the compression of the classical bit count which enables our quantum algorithm for exponential speedup over classical algorithms, as shown below in a recursive computation on an 8 classical bit system.

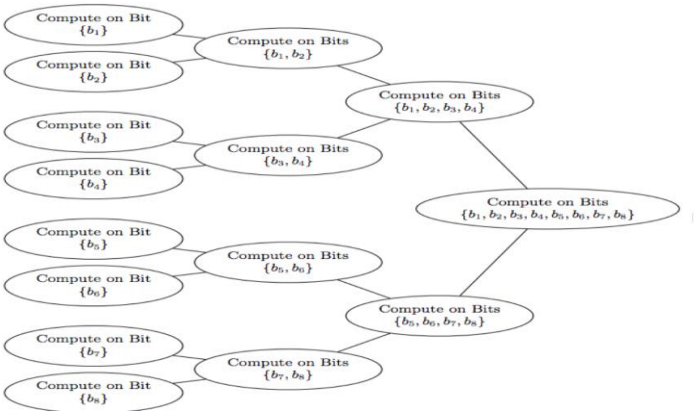


**Fig. 14.** Packing bits [67].

Algebraically we may represent the loading of four classical bits into a state of 3 entangled quantum bits as:

$$|\psi_A\rangle = |\psi_{\alpha\beta\gamma}\rangle = |\alpha\beta\gamma\rangle = |00\rangle \otimes |b_{00}\rangle + |01\rangle \otimes |b_{01}\rangle + |10\rangle \otimes |b_{10}\rangle + |11\rangle \otimes |b_{11}\rangle ,$$
$$\equiv |00b_{00}\rangle + |01b_{01}\rangle + |10b_{10}\rangle + |11b_{11}\rangle, \qquad (3.1.1)$$

where A is a matrix containing 4 classical bit values of $\{b_{00}, b_{01}, b_{10}, b_{11}\}$ which are loaded into the three qubits $\{\alpha, \beta, \gamma\}$. We must also optimize the data transfer from the classical domain into the quantum domain. To do so, we recall two results: [42], [44] Shannon's Capacity Theorem and the Coding Theorem. The first provides an upper bound on the rate at which information can be transmitted, and the second states that as long as the information rate is less than or equal to the channel capacity, then there is a coding technique such that information can be transmitted over the channel with an arbitrarily small probability of error. If the information rate exceeds the channel capacity, then error-free transmission is impossible. [45] Thus, these two results dictate that we must compress the incoming bitstream in the

classical domain, pass it through the channel, and then decompress the data stream at the output in the quantum domain.

The approach to compress the classical input data is as follows: suppose we have a block of classical data of length N bits. We then compress the N bits by a factor of $L$, where $L = -E_S[log(p_i)]$ is the average entropy of a bit in the incoming bitstream. If the length $N \gg 1$, then a single bit is mapped to $L$, for $0 \leq L \leq 1$. Thus, the block of length $N$ is mapped to $M = LN$bits such that $0 \leq M \leq N$.The compression and decompression methods improve the data transfer by virtue of the number of stages required in the circuit. This, in turn, reduces the number of loading circuits that are needed. The particular type of circuit we are interested in has similarities to Fig. 13 (Cortese and Braje, 2018). When implemented into a quantum algorithm, this circuit exhibits an exponential speedup compared to its classical counterpart. Furthermore, we must show that the circuit's time complexity is $O(log(N))$.

### 3.1.3 Properties of quantum channels

We now acknowledge some of the properties of quantum channels. For our purposes, we assume that all state spaces are finite-dimensional. There are three ways we must look at this: the first is from the Schrodinger perspective, which makes use of density matrices acting on the relevant state-spaces; the second is from the Heisenberg perspective, which extends density matrices acting on $H_A$ to the full space of operators, and the third from the classical point of view, to account for the cases in which the input data are classical in nature [38], [45].

In the case of Schrodinger, let $H_A$ and $H_B$be Hilbert spaces of dimensions $n$ and $m$, respectively. Also, let $L(H_A)$denote the family of operators acting on $H_A$, and let $L(H_B)$be defined similarly on $H_B$. Then we define a purely quantum channel to be a mapping $\Phi$: $\rho(H_A) \to \rho(H_B)$between density matrices acting on the Hilbert spaces $H_A$and $H_B$such that $\Phi$is linear, $\Phi$is a positive map, the induced map, $I_n \otimes \Phi$,is a completely positive map, and that for all density matrices $\rho, tr(\rho) = 1$.This then implies that $\Phi$must preserve traces [44].

Since density matrices are only one type of operator, when viewing quantum channels in the context of the Heisenberg picture, they form a (proper) subset of the operators acting on $H_A$. Therefore, in this perspective, we extend density matrices to the entire space of operators. We can do this once we have established a mapping between density matrices and make use of the linearity property and the finite-dimensional assumption made above. Thus, let us consider $L(H_A)$and $L(H_B)$and the map $\Phi$: $L(H_A) \to L(H_B)$ [42]. Note that the spaces of operators are Hilbert spaces equipped with the Hilbert-Schmidt inner product. Therefore, we may obtain the adjoint map $\Phi^*$defined by the rule $\langle A, \Phi(\rho)\rangle = \langle \Phi^*(A), \rho\rangle$. This map takes observables on $H_B$to observables to $H_A$. Additionally, we can check that the adjoint map $\Phi^*$is unital; that is, $\Phi^*(I) = I$ provided that $\Phi$is trace-preserving.

So far, we have only considered cases for which the input data is quantum. However, we must also consider the case where the input data is classical. Therefore, we must generalize this treatment further in order to account for this. Thus, let $\Psi$: $L(H_B) \to L(H_A)$ be a linear map between the spaces of operators such that $\Psi$is unital, and is a completely positive map. Viewing the spaces of operators as $C^*$-algebras we can then we can redefine our linear map to be an unital, completely positive map between $C^*$-algebras. Consequently, we can append the classical input to the operator space

under the tensor product via $\Psi: L(H_B) \otimes C(X) \to L(H_A)$, where $C(X)$ is the space of continuous functions defined on the set $X$.

Some examples of quantum channels include states, observables, and a measure-and-prepare channel we now consider. Let us suppose that party A and party B wish to send information between them. [15][17][37] Party A measures an observable and sends the result to B classically. B then prepares their quantum system. In the Schrodinger perspective, the measure-and-prepare channel is given by the composition $\Phi(\rho) = (\Phi_2 \circ \Phi_1)(\rho) = \sum_i \quad \rho(F_i)R_i$, where $F_i$ and $R_i$denote the $i$-th quantum state, and $\Phi_1$and $\Phi_2$denote the measurement map and preparation map, respectively.

### 3.1.4 Quantum mechanism for zero-knowledge proof

In cryptography, the zero-knowledge protocol is a method in which one party conveys to another party, some verifier, that they know some value without conveying any information apart from the fact that they have that value. The essence of zero-knowledge proofs is to prove that one possesses such information without revealing the information itself or any additional information. These are done by trapdoor functions (like hash algorithms like SHA-256) [1], [6], [51], that can only be inverted by knowing the solution or searching the entirety of the domain, which is an NP-hard problem. However, using quantum algorithms, one could search the function's domain to effectively tamper or find out the original information in $O(n)$. On the other hand, if one were to attack a user in a blockchain application that transmits classical data into a quantum channel, they would find it difficult to copy data encoded in a quantum state [31], [44]. This means that if one tried to observe or eavesdrop on the quantum state, the qubit state would be changed due to wave function collapse (no-cloning theorem), meaning that the server will be indicated if a user attempts to tamper with data. [38].

We may prove that two people have the exact quantum representation of the data (near 0 chance of collisions and near 0 probability of someone finding a collision as qubits lose their coherence every time they are measured) if $|(Pr\ (|\psi\rangle\ ) = 1) - (Pr\ (|\psi'\rangle\ = 1))| < \varepsilon$, such that $|\psi\rangle$ and $|\psi'\rangle$ representing two quantum states of two separately entered data set and where $\varepsilon$is negligible.

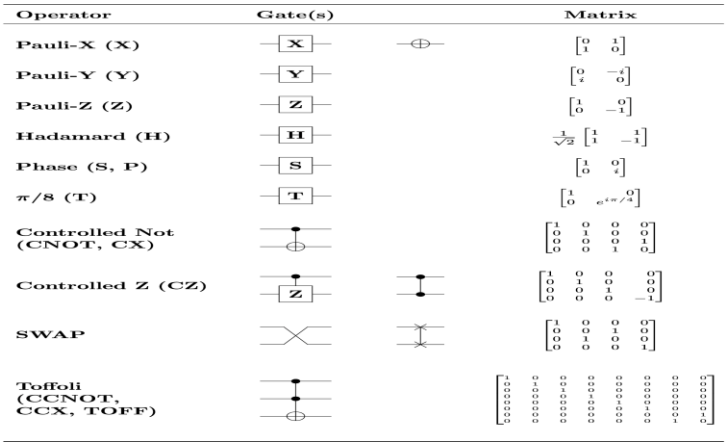


**Fig. 15.** Diagrams of basic quantum gates [68].

Implementing packet quantum network intercommunication between multiple parties (for example, swap gates between two qubits to swap quantum states in a circuit) would enable us to transmit any zero-knowledge proof where some user data exists and has associated data information.

In numerous large-scale blockchains, application servers use the existence hash trees or Merkle trees in which every leaf node is labeled with the cryptographic hash of a data block, and every non-leaf node is labeled with the cryptographic hash of the labels of its child nodes. The implementation is with the objective for storage optimization while also retaining the zero-knowledge proof that some data value exists [6], [53]. We may similarly use the entanglement of qubit states (in a tree-like structure) to retain zero-knowledge proof to entangle the quantum representation of numerous data blocks while retaining the timestamp and block information [1], [37].

## 3.2 Asymmetric Encryption Algorithm in the Fresnel Domain

### 3.2.1 Intuition

In blockchain applications, the authenticity and integrity of a message, software, or digital document are validated using digital signature technologies, with the primary implementation being the RSA encryption algorithm. Encryption and decryption typically work using encryption algorithms and mathematically generated keys to turn plaintext to ciphertext and back to plaintext. [11] [49]. However, because of the nature of quantum computing algorithms, such as Shor's algorithm, many fundamental encryption algorithms are not quantum resistant [6]. We, therefore, propose an optical scheme for information encryption under a double random phase encoding framework using phase-truncation in the Fresnel domain with enhanced complexity and immunity.

To increase the security of the double random phase encoding, numerous image encryption schemes (such as fractal Fourier transform, the gyrator transform, Fraunhofer domain, fresnel, etc.) [19], [27] have been proposed to transform our double random phase encoding in different domains for placing encryption keys. We use the fresnel transform to improve the robustness as we are offered numerous keys, such as the wavelength, free space propagation distance, and sampling parameters. The Fresnel transform is also aberration-free [34] and easier to implement digitally [46].

In this research, we use a plaintext encryption scheme using a fresnel domain phase encoding and a modified Gerchberg– Saxton phase retrieval algorithm [46]. The encryption scheme for an image is done by converting the primary image into a phase-only mask which is then used as a fresnel domain key for encrypting further data and a random intensity image (RIM) using random phase masks. We further add robustness by generating an asymmetric key from the generated random intensity image [19], [20], [34]. To decrypt the primary image, we use a known-plaintext attack method. In order to find the original image, one must figure out the phase-only mask, which requires the use of the following encryption keys: the random intensity image (which can only be found out by finding the phase truncated value and amplitude truncated value of the random intensity image), the random phase mask, and the encrypted image.

### 3.2.2 Single color encryption scheme

The steps to a single color encryption scheme are as follows:

Suppose that f(x,y) is the primary single color image we use to input into the encryption algorithm. In this research, to create double phase encoding in the Fresnel domain, we have to initially apply a (G-S) phase retrieval algorithm (an interactive phase retrieval algorithm to find the phase component of a complex wavefront with the goal of minimal error) [46]. The steps for the Gerchberg–Saxton algorithm are as follows:

We first separate the phase and amplitude of an image by multiplying the amplitude of the original image by a random phase mask then taking its fresnel transform [35].

$$f'_1(x,y) = |f(x,y)| * exp\{i2\pi r_1(x,y)\},\ f'_n(x,y) = |f(x,y)| * exp\{i2\pi r_n\ (x,y) \tag{3.2.1}$$

$$F_{n+1}(u,v) = FrT^{\ z}_{\ \lambda}[f'_n(x,y)] = \frac{exp\{\frac{i2\pi z}{\lambda}\}}{\sqrt{i\lambda z}} \int \int f'_n(x,y) * exp\left[\frac{i\pi}{\lambda z}((x-u\ )^2 + (y-v)^2)\right] dxdy = |F_{n+1}(u,v)| * exp\{i\varphi_n(u,v\} \tag{3.2.2}$$

Where $f'_n$denotes n iterations through the G-S phase retrieval algorithm, z is the propagation distance, and $\lambda$ is the wavelength of the transform. Supplementary (x, y) and (u, v) represent the input and output domain of the function respectively.

We then replace the amplitude of $F_{n+1}(u,v)$,as shown: $F'_{n+1}(u,v) = 1 * exp\{i\varphi_n(u,v)\}$
Now, we apply the inverse Fresnel transform $F'_{n+1}(u,v)$ as:

$$F''_{n+1}(x,y) = FrT_{\lambda}^{-z}[F'_{n+1}(u,v)] = |F''_{n+1}(x,y)| * exp\{i\varphi'_n(x,y)\} \tag{3.2.3}$$

We get that

$$f'_n(x,y) = |f(x,y)| * exp\{i\varphi'_n(u,v)\} = exp\{ir_{n+1}(u,v)\} \tag{3.2.4}$$

The final iteration of the modified G-S phase algorithm is completed by computing the mean-square error between $abs[f'_n(x,y)]$and $abs[f(x,y)]$

$$MSE = \frac{\sum_{x=0}^{N-1}\sum_{x=0}^{N-1}\left\{|f'_n(x,y| - |f(x,y}\right\}|\ \}^2}{N^2}. \tag{3.2.5}$$

Once we have obtained the phase-only mask, we use the resultant phase-only mask as a Fresnel domain key for encrypting randomly generated data, which is done by the following steps:

We first encrypt some random intensity mask denoted as $e(x,y)$,which is then bonded with a random phase mask ( $exp\{i2\pi R(x,y)\}$ )whose Fresnel transform is calculated. Finally, we multiply by the already generated phase-only mask, and its Fresnel transform is calculated again to generate our encrypted image [34], [35].

$$E_1(u,v) = FrT^{z_1}_{\lambda}\ [e(x,y) * i2\pi R(x,y)] \quad (3.2.6)$$

$$E(x,y) = FrT^{z_2}_{\lambda}\ [E_1(u,v) * exp\{i\phi_n(u,v)] \quad (3.2.7)$$

In order to further enhance security, we can make our random intensity image generate two asymmetric keys. [20][21][39] This is done by taking the Fresnel transform of the product between a statistically independent random phase mask $exp\{i2\pi R_1(x,y)\}$and the random intensity image, which are then separated as two asymmetric keys (one key being the spectrum's phase-truncation and the spectrum's amplitude- truncation).

$$e_1(u,v) = PT\{\ FrT^{z_3}_{\lambda}\ [e(x,y) * exp\{i2\pi R_1(x,y)\}\ ] \quad (3.2.8)$$

$$p_1(u,v) = AT\{\ FrT^{z_3}_{\lambda}\ [e(x,y) * exp\{i2\pi R_1(x,y)\}\ ] \quad (3.2.9)$$

### 3.2.3 Multi-color encryption scheme

The steps for a multicolor encryption scheme is as follows:

Suppose we have a color image f(x,y) which can transcript as three color channels $f_r(x,y)$, $f_g(x,y)$, and $f_b(x,y)$[25]. We use the process used in the previous section to convert each color component into phase-only masks.

$$F_{n+1\{red\}}(u,v) = FrT_z^{\lambda}\ [f'_{n\{red\}}(x,y)] = |F_{n+1\{red\}}(u,v)| * exp\{i\varphi_{nr}(u,v\}\} \quad (3.2.10)$$

$$F_{n+1\{green\}}(u,v) = FrT_z^{\lambda}\ [f'_{n\{green\}}(x,y)] = |F_{n+1\{green\}}(u,v)| * exp\{i\varphi_{ng}(u,v\}\} \quad (3.2.11)$$

$$F_{n+1\{blue\}}(u,v) = FrT_z^{\lambda}\ [f'_{n\{blue\}}(x,y)] = |F_{n+1\{blue\}}(u,v)| * exp\{i\varphi_{nb}(u,v\}\} \quad (3.2.12)$$

The color components $f_r(x,y), f_g(x,y), f_b(x,y)$ are embedded with random intensity images $e_r(x,y), e_g(x,y), e_b(x,y)$, as shown:

$$E_{1R} = FrT_{\lambda}^{z_1}[e_r(x,y) * exp\{i2\pi R_R(x,y)\}], E_R = FrT_{\lambda}^{z_2}[E_{1r}(u,v) * exp\{i\varphi_{nr}(u,v)\}] \quad (3.2.13)$$

$$E_{1G} = FrT_{\lambda}^{z_1}[e_G(x,y) * exp\{i2\pi R_G(x,y)\}], E_G = FrT_{\lambda}^{z_2}[E_{1G}(u,v) * exp\{i\varphi_{ng}(u,v)\}] \quad (3.2.14)$$

$$E_{1B} = FrT_{\lambda}^{z_1}[e_B(x,y) * exp\{i2\pi R_B(x,y)\}], E_B = FrT_{\lambda}^{z_2}[E_{1B}(u,v) * exp\{i\varphi_{nb}(u,v)\}] \quad (3.2.15)$$

With the three color asymmetric keys generated as follows:

$$e_{1r}(u,v) = PT\{\ FrT^{z_3}_{\lambda}\ [e_r(x,y) * exp\{i2\pi R_r(x,y)\}\ ] \quad (3.2.16)$$

$$p_{1r}(u,v) = AT\{\ FrT^{z_3}_{\lambda}\ [e_r(x,y) * exp\{i2\pi R_r(x,y)\}\ ]$$

$$e_{1g}(u,v) = PT\{\ FrT^{z_3}_{\lambda}\ [e_g(x,y) * exp\{i2\pi R_g(x,y)\}\ ] \quad (3.2.17)$$

$$p_{1g}(u,v) = AT\{ FrT^{z_3}_{\lambda} [e_g(x,y) * exp\{i2\pi R_g(x,y)\}]$$

$$e_{1b}(u,v) = PT\{ FrT^{z_3}_{\lambda} [e_b(x,y) * exp\{i2\pi R_b(x,y)\}] \quad (3.2.18)$$

$$p_{1b}(u,v) = AT\{ FrT^{z_3}_{\lambda} [e_b(x,y) * exp\{i2\pi R_b(x,y)\}]$$

### 3.2.4 Single color decryption scheme

The steps to a single color decryption scheme are as follows:

The goal of decryption is to figure out the original single color image, which is done by finding out the random phase mask used. We find out the phase-only mask by figuring out the following: the encrypted image, the propagation distances, the random intensity image (which is only found when a user has both asymmetric keys, those being the amplitude truncated value and phase truncated value of the RIM), and the already generated random phase mask. Thus, the following steps find the original single color image [20], [25], [34]:

$$e(x,y) = PT\{ FrT_{\lambda}^{-z_3}[e_1(u,v) * p_1(u,v)]\} \quad (3.2.19)$$

$$exp\{i\varphi_n(u,v)\} = \frac{FrT_{\lambda}^{-z_2}[E(x,y)]}{FrT_{\lambda}^{-z_1}[|e(x,y)| * exp\{i2\pi R(x,y)\}]} \quad (3.2.20)$$

$$f(x,y) = FrT_{\lambda}^{-z}[exp\{i\varphi_n(u,v)\}] \quad (3.2.21)$$

The specific implementation of the encryption will be further specified in 4.3.

### 3.2.5 Multi-color decryption scheme

The steps to a multicolor decryption scheme are as follows:

We must first find each color channel for the original image [19], which is done by finding out each color channel's random phase mask used. Then, we find out each component's phase-only mask by figuring out the following: the encrypted color channel, the propagation distances used, each color channel's random intensity image (which is only found when a user has both asymmetric keys, those being the amplitude truncated value and phase truncated value of the RIM), and each color channel's already generated random phase mask [22], [25]. Thus, the original color image can be found by the following steps:

$$e_r(x,y) = PT\{ FrT_{\lambda}^{-z_3}[e_{1r}(u,v) * p_{1r}(u,v)]\} \quad (3.2.22)$$

$$exp\{i\varphi_{nr}(u,v)\} = \frac{FrT_{\lambda}^{-z_2}[E_r(x,y)]}{FrT_{\lambda}^{-z_1}[|e_r(x,y)| * exp\{i2\pi R_r(x,y)\}]} \quad (3.2.23)$$

$$f_r(x,y) = FrT_{\lambda}^{-z}[exp\{i\varphi_{nr}(u,v)\} \quad (3.2.24)$$

$$e_g(x,y) = PT\{ FrT_{\lambda}^{-z_3}[e_{1g}(u,v) * p_{1g}(u,v)]\} \quad (3.2.25)$$

$$exp\{i\varphi_{ng}(u,v)\} = \frac{FrT_{\lambda}^{-z_2}[E_g(x,y)]}{FrT_{\lambda}^{-z_1}[|e_g(x,y)| * exp\{i2\pi R_g(x,y)\}]} \quad (3.2.26)$$

$$f(x,y) = FrT_{\lambda}^{-z}[exp\{i\varphi_{ng}(u,v)\} \quad (3.2.27)$$

$$e_b(x,y) = PT\{\, FrT_{\lambda}^{-z_3}[e_{1b}(u,v) * p_{1b}(u,v)\,]\} \quad (3.2.28)$$

$$exp\{i\varphi_{nb}(u,v)\} = \frac{FrT_{\lambda}^{-z_2}[E_b(x,y)]}{FrT_{\lambda}^{-z_1}[|e_b(x,y)| * exp\{i2\pi R_b(x,y)\}]} \quad (3.2.29)$$

$$f_b(x,y) = FrT_{\lambda}^{-z}[exp\{i\varphi_{nb}(u,v)\} \quad (3.2.30)$$

$$f(x,y) = f_r(x,y) + f_g(x,y) + f_b(x,y) \quad (3.2.31)$$

Further security measures may be taken by applying an m-band discrete wavelet transform then creating a multi-image encryption scheme that would have been decrypted [34] then combined with another through an inverse m-band wavelet transform. This paper proposes additional encryption security through pseudo-quantum image steganography, though there are innumerable post-quantum security schemes [22].

### 3.3 Color QR Code Using a Pseudo-Quantum Steganography Mechanism

In this research, we create a color QR code that uses pseudo quantum steganography to store more information than a standard QR code and embed additional data with limited access privileges.

#### 3.3.1 Connecting quantum and pseudo-quantum steganography

When one has qubit in superposition $S_{ij}$ such that, $|S_{ij}\rangle = a_{ij}|0\rangle + b_{ij}|1\rangle$, one could embed quantum information into approximation coefficients of a wavelet transformed data using a quantum computer as shown:

$$\theta_{ij}^{E} = \begin{cases} \cos^{-1}\left(\cos(\theta_{ij}) + \delta\cos(x_{ij})\right) & \text{if } |P_{ij_1}| \geq |P_{ij_2}| \\ \sin^{-1}\left(\sin(\theta_{ij}) + \delta\sin(x_{ij})\right) & \text{if } |P_{ij_1}| < |P_{ij_2}| \end{cases} \quad (3.3.1)$$

However, if one does not have access to a quantum computer, the following steps are used to simulate a pseudo-quantum steganography process [2], [3], [22].

#### 3.3.2 Steps to pseudo-quantum embedding

When embedding information once a stacked RGB-color qr is generated, $f_r(x,y) + f_g(x,y) + f_b(x,y) = f(x,y)$, we can then start the embedding process for which we split our color QR into 3 color channels which we can then embed additional secret information using pseudo-quantum steganography [22].

Step 1: Wavelet Transform: We perform wavelet transform on each channel of the color barcode to obtain the approximation portion: $T = WIW^t$, where T is the wavelet transformed barcode I, and W is the wavelet transform matrix [6].

Step 2: Embedding process: Because of lack of access to a quantum computer one can mimic a quantum bit with the use of Pseudo Quantum Signals. In order to perform this method, we first transform the approximation image and the secret information into angle signals using the following linear transformations:

$$\theta_{ij} = \frac{\pi(A_{ij} + \mu_1 - 2\mu_2)}{6(\mu_1 - \mu_2)} \text{ where } \mu_1 = max(A_{ij}), and\ \mu_2 = min(A_{ij}) \quad (3.3.2)$$

$$\alpha_{ij} = \frac{\pi(X_{ij} + v_1 - 2v_2)}{6(v_1 - v_2)} \text{ where } v_1 = max(X_{ij}), and\ v_2 = min(X_{ij}), \quad (3.3.3)$$

Such that $\theta_{ij}$ is the pseudo quantum signal angle vector of the wavelet approximation matrix $A$ and $\alpha_{ij}$ is the angle signal vector of some secret information X. [2] The general linear transform ensures that the new approximation coefficients $\theta_{ij}, \alpha_{ij}$ are bounded by $[\frac{\pi}{6}, \frac{\pi}{3}]$.

We can further use the uncertainty principle of quantum bits, to create a matrix K using a random number generator such that the values of $k_{ij}$ are bounded by [0,1]. We can then perform the Pseudo Quantum Signal Embedding as shown:

$$\theta_{E,ij} = m\begin{cases} \cos^{-1}(\cos(\frac{\theta_{ij}}{m}) + \varepsilon\cos(\frac{\alpha_{ij}}{n})), & k_{ij} \geq 0.5 \\ \sin^{-1}(\sin(\frac{\theta_{ij}}{m}) + \varepsilon\sin(\frac{\alpha_{ij}}{n})), & k_{ij} < 0.5 \end{cases}, \quad (3.3.4)$$

where m,n are 1 and $\varepsilon$ is our embedding intensity.

Once the embedding process is done, we can perform the inverse of the linear transform to obtain the embedded approximation wavelet as shown

$$A_{E,ij} = \frac{6\theta_{e,ij}(\mu_1 - \mu_2)}{\pi} - \mu_1 + 2v_1 \quad (3.3.5)$$

Step 3 Inverse Wavelet transform: We finally replace the embedded approximation portion of the wavelet transformed matrix, $T_E$, to get the embedded color QR, $I_E$ , as shown.

### 3.3.2 Extracting data from pseudo-quantum signals

Step 1 obtain pseudo-quantum signals: To generate the pseudo-quantum signals, one must copy the original and embedded barcode. [22][23] Then after finding the approximation space of each matrix in the wavelet domain and applying the linear pseudo quantum signal transform to both barcodes, we obtain $\theta$ and $\theta_E$.

$$T = WIW^{-1} \quad (3.3.6)$$

$$T_E = WIW^{-1} \quad (3.3.7)$$

$$\theta_{ij} = \frac{\pi(A_{ij} + \mu_1 - 2\mu_2)}{6(\mu_1 - \mu_2)}, \text{ where } \mu_1 = max(A_{ij}), and\ \mu_2 = min(A_{ij}) \quad (3.3.8)$$

$$\theta_{E,ij} = \frac{\pi(A_{E,ij} + v_1 - 2v_2)}{6(v_1 - v_2)}, \text{ where } v_1 = max(A_{E,ij}), and\ v_2 = min(A_{E,ij}) \quad (3.3.9)$$

Step 2 extracting secret signals: Given signals $\theta, \theta_E$, the extraction process for signal $\alpha$ is as follows:

$$\alpha_{ij} = n\begin{cases} \cos^{-1}\left(\dfrac{\cos\left(\frac{\theta_{E,ij}}{m}\right) - \cos\left(\frac{\theta_{ij}}{m}\right)}{\varepsilon}\right), & k_{ij} \geq 0.5 \\ \sin^{-1}\left(\dfrac{\sin\left(\frac{\theta_{E,ij}}{m}\right) - \sin\left(\frac{\theta_{ij}}{m}\right)}{\varepsilon}\right), & k_{ij} < 0.5 \end{cases} \quad (3.3.10)$$

where m, n, are 1.

Step 3 Obtaining Secret Information: In order to obtain the secret, we use the inverse of a linear transformation, which was performed during the encrypting procedure as shown [7]:

$$SecrectData_{ij} = \frac{6\alpha_{ij}(v_1 - v_2)}{\pi} - v_1 + 2v_2 \quad (3.3.11)$$

Confidential data may include audio files, an image, numeric information, even text information, etc.

## 4. Blockchain Application and Experiment Results

### 4.1 Datasets

We will use an auto-generated dataset and random images, among other numeric arrays, to embed in our research to simulate our mechanisms in MATLAB and python environments. The dataset consists of a group of person objects that takes any number of arguments so long as one enters a tested characteristic and the corresponding data to the information (the method is modified to take string, numeric, float, timestamps, among other data types). However, for the sake of simulating the experiment results, the dataset will generate a list of random people objects that generate the following medical characteristic and corresponding data values: Last visited year (random [2000 to 2021] ), Medical Center ID ( random [0, 50]), timestamp of appointment (random [2021 January 1st 0:00 am to 2021 December 31st 11:59 pm]), age (random [0, 121]), weight (random [5, 300]), zip code (random[00501, 99950]), number of covid shots (random [0, 2]), number of tetanus shots (random [0, 2]), bmi (random [0, 40]), diabetes one test (random [YES, NO]), diabetes two test (random [YES, NO]), allergies (random [YES, NO]), if a patient is working (random [YES, NO]), prior

mental health conditions (random [YES, NO]), heart rate (random [YES, NO]), and whether or not a patient needs further medical assistance (random [YES, NO]) for experimentation's sake.

The dataset will be tested, sent back and forth, embedded, extracted, encrypted, decrypted, and chaos tested to simulate the proposed privacy-preserving post-quantum blockchain mechanisms.

## 4.2 Post-Quantum Blockchain Design Using Pseudo Quantum Signal

One of the exciting applications of quantum computers is Grover's algorithm. Classically, searching an unsorted database for a fixed-length output requires a linear search in $O(n)$ time. Grover's algorithm advantages quantum mechanics to$O(\sqrt{n})$ which [6] is a significant speedup for even small domains, which poses a great threat to cryptographic systems such as in the use of hashing in blockchain applications for zero-knowledge proof of the existence of data.[10][18] However, if one were to take advantage of the use of classical to quantum mappings for zero-knowledge proof and quantum decoherence (the loss of quantum coherence/information), an attacker would find that an observed quantum state would lose information far too expo distally to find a classical mapping to the original data. Quantum decoherence has been used to understand the possibility of the collapse of the wave function in quantum mechanics [43], [44]. Decoherence does not generate actual wave-function collapse. It only provides a framework for apparent wave-function collapse, as the quantum nature of the system "leaks'' into the environment [53]. Meaning that the wave function of components is decoupled from a coherent quantum state and acquires different phases from their immediate surroundings. Because of this property, users in our post-blockchain application are left with a near-zero probability that an attacker would be able to figure out their classical mapping from a qubit state but ensured the existence of a relatively straightforward theoretical classical to a quantum channel for zero-knowledge proof of work.

### 4.2.1 Data-cleaning and conversion

In this research, we lack access to a quantum computer and an inability to transmit data through classical to quantum mapping, [42],[43],[44] but we can simulate a post-quantum blockchain structure using pseudo-quantum signals [2], [3], [22]. The steps used in a classical mapping to pseudo quantum mapping are as follows:

We separate our generic person input into two raw arrays of qualitative and quantitative data to turn the associated quantitative data into a quantum/pseudo quantum state. In the pseudo quantum channel, we convert any data into numerical data. Data that was already identified as a float value remained; any binary string data such 'yes' or 'no' were also numerically converted to simply 1's and 0's, and finally, any other such data (namely strings) was converted into there hexadecimal ASCII values then converted numerically base 16.

Further data optimization could be done by taking the scientific notation of large values however, for the sake of data collisions, data loss from the rounding error, and simplicity of data that will not be done; however one could decompose substantial numerical values into an array of numerics as shown:

'Hello world' → $48656c6c6f20776f726c64$ → 875216180888826712310 69284 → 87521 | 61808 | 88826 | 712310 | 69284

The final step to converting the numeric data into a pseudo quantum signal is performed:

$$\theta_{ij} = \frac{\pi(A_{ij} + \mu_1 - 2\mu_2)}{6(\mu_1 - \mu_2)} \text{ Where } \mu_1 = max(A_{ij}), and\ \mu_2 = min(A_{ij}) \quad (4.2.1)$$

The following shows the generation of a random person object with associated data. The data is processed into a numeric set than to a pseudo quantum signal and finally transmitted over a network via its byte array representation, and:

Visit time | PatientID | Name | Age | Weight | Zipcode
| Number of covid shots | Number of tetanus shots | bmi | diabetes1test
| diabetes2test | Allergies | isWorking | mental health condition | heart rate | needs further medical assistance

| 20211128045305 | 88485 | Sylvia Hemingway | 15 | 328 | 38351 | 1 | 0 |
3.11 | no | yes | yes | yes | no | 85 | yes

| 2.021113 | 88485.0 | 7.033548 | 15.0 | 328.0 | 38351.0 | 1.0 | 0.0 | 3.11 | 0.0
| 0 | 1.0 | 1.0 | 1.0 | 0.0 | 85.0 | 1.0
[0.52361074 1.04719755 0.5236404 0.52368754 0.52553967 0.75053596
0.52360469 0.52359878 0.52361718 0.52359878 0.52360469 0.52360469
0.52360469 0.52359878 0.52410175 0.52360469]

b'\x80\x02cnumpy.core.multiarray\n_reconstruct\nq\x00cnumpy\nndarray\nq\x01K\x00\x85q\x02c_codecs\nencode\nq\x03X\x01\x00\x00\x00bq\x04X\x06\x00\x00\x00latin1q\x05\x86q\x06Rq\x07\x87q\x08Rq\t(K\x01K\x10\x85q\ncnumpy\ndtype\nq\x0bX\x02\x00\x00\x00f8q\x0c\x89\x88\x87q\rRq\x0e(K\x03X\x01\x00\x00\x00<q\x0fNNNJ\xff\xff\xff\xffJ\xff\xff\xff\xffK\x00tq\x10b\x89h\x03X\xb7\x00\x00\x00\xc3\x8e4\xc3\xbbLk\xc3\x81\xc3\xa0?es-
8R\xc3\x81\xc3\xb0?\xc2\xa6\xc2\xa7\xc3\x90\xc2\x80\xc2\xa9\xc3\x81\xc3\xa0?\xc3\xb3U)]\x0c\xc3\x82\xc3\xa0?I\xc2\xbc\x17\xc2\x948\xc3\x91\xc3\xa0?\xc2\x8b\xc3\xb4\xc2\x9c\xc3\xbcc\x04\xc3\xa8?\x08\x0b\x0b\xc2\xa1^\xc3\x81\xc3\xa0?es-8R\xc3\x81\xc3\xa0?N\xc3\x83:\xc3\x90x\xc3\x81\xc3\xa0?es-
8R\xc3\x81\xc3\xa0?\x08\x0b\x0b\xc2\xa1^\xc3\x81\xc3\xa0?\x08\x0b\x0b\xc2\xa1^\xc3\x81\xc3\xa0?\x08\x0b\x0b\xc2\xa1^\xc3\x81\xc3\xa0?es-
8R\xc3\x81\xc3\xa0?\xc2\x87\xc3\x8c\xc3\x80\tq\xc3\x85\xc3\xa0?\x08\x0b\x0b\xc2\xa1^\xc3\x81\xc3\xa0?q\x11h\x05\x86q\x12Rq\x13tq\x14b.'

[0.52361074 1.04719755 0.5236404 0.52368754 0.52553967 0.75053596
0.52360469 0.52359878 0.52361718 0.52359878 0.52360469 0.52360469
0.52360469 0.52359878 0.52410175 0.52360469]

#### 4.2.2 P2P network

Once we have acquired a quantum signal (or pseudo quantum signal), we must create a linked list of data blocks that are maintained and accessible to all users in the network. Each block contains a timestamp, a quantum (or pseudo quantum) representation of the data, and a unique tag identifier.

The protocol for packet quantum network intercommunication between multiple parties is a topic of active research. In such a quantum network for a blockchain application, the quantum signal representation of the data will be generated by the user several times [39] (the amount required to send to all the other peers in the network because of the no-cloning rule) and then sent to all peers in the network; in which the network will attach a timestamp and a data tag for verified blocks [48].

Classically, we implemented a simple peer to peer network by creating a centralized server and used socket programming to communicate from a client peer to all other peers via a centralized server; however, in an actual blockchain application, a centralized server would not exist, but this was sufficient to test the central concept of our algorithm. A socket can be used to send arbitrary numbers of bytes typically called payload over a network [48].

In our experiments, we first converted the data into a pseudo quantum signal ( represented by a NumPy array in python) which is then converted to a byte array for sending over a socket. Then, the byte array was converted back to the data's pseudo quantum signal representation on the receiving side.

### 4.3 Color QR Code Embedding Process

Steganography is the practice of concealing a message within another message or physical image often concealed by a computer file, message, image, or video. The advantage of messages that are transmitted using steganography is that the intended secret message does not attract attention to itself. Because of their nature, color QRs lend themselves to a valuable key to steganography-based applications. In our paper, we use an RGB-color QR code that stores more information than a standard QR code, with the ability to embed extra data with limited access privileges and with the same readability as the unembedded image. [2][22] We show how we can embed and extract both files and images into a color QR code [23] using pseudo-quantum steganography with limited accessibility and complete concealment for the intent of digital signature encryption information.

#### 4.3.1 Creation and embedding

We first generate three QR codes that individually encode data, then uniquely color each QR. For example, as shown below, we generate a red color QR code encoding a name, a green color QR encoding a patient ID, and a blue color QR encoding date of birth.

```
qrRed.add_data('NAME:SONOK')
qrGreen.add_data('PATIENTID:784901')
qrBlue.add_data('DATE OF BIRTH: JAN 21 2005')
```

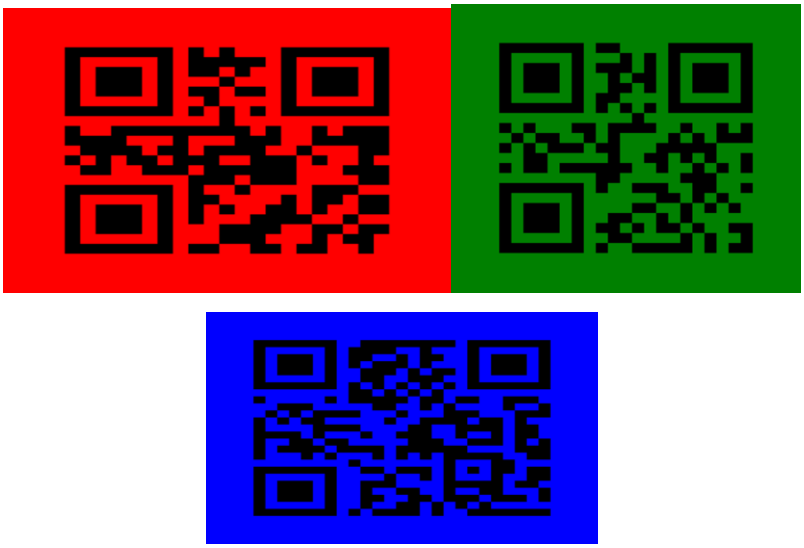

**Fig. 16.** Color QRs encoding data.

Once we have obtained three separate color QRs, we can merge the 3 QRs into an RGB color QR code. This is done by taking the red color channel of the red-colored QR, the blue color channel of the blue-colored QR, and the green channel of the green-colored QR, then merging to form a three-dimensional color QR image, and now we can start the pseudo-quantum steganography process [22].

```
\img = Image.merge('RGB', (redChannelOfRedQR,greenChannelOfGreenQR,
blueChannelOfBlueQR)) # PYTHON ENVIRONMENT

img.imshow()

watermark.imshow()
```

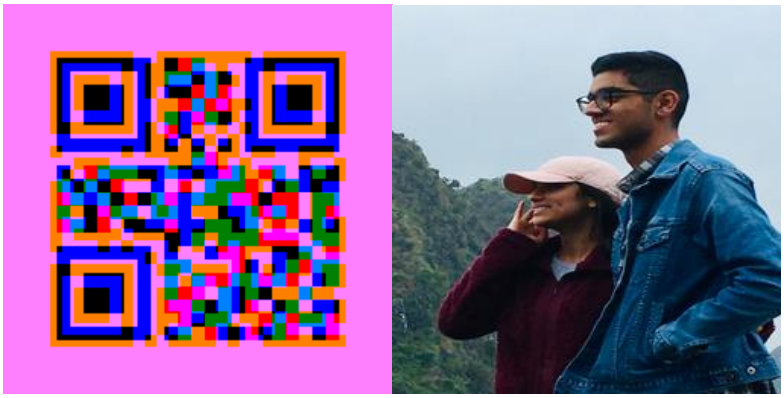

**Fig. 17.** Watermark and unwatermarked RGB color QR.

The following image is the watermarked color QR code. Because of the low embedding intensity, a color QR reader will read the watermarked image as if it was an unwatermarked image.

```
imshow(waterMarkedImageE1) % MATLAB CODE, embedding intensity e = .001
imshow(waterMarkedImageE2) % MATLAB CODE, embedding intensity e = .01
imshow(waterMarkedImageE3) % MATLAB CODE, embedding intensity e = .1
```

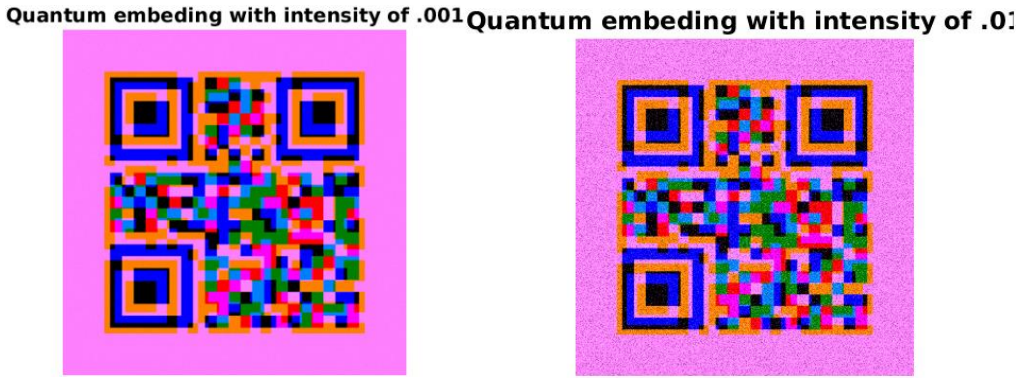


**Fig. 18.** Embedded images with intensity of .001 and.01.

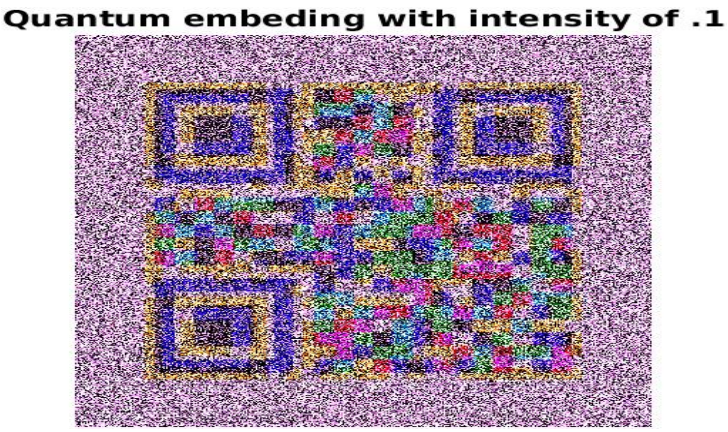


**Fig. 19.** Embedded images with intensity of .1.

Estimation of the embedding intensity $\varepsilon$: Since $cos^{-1}$has a domain [0,1] according to our pseudo-quantum embedding scheme $cos\theta_{w,mn} = cos\theta_{mn} + \varepsilon * cos\ \alpha_{mn} \leq 1$ meaning$\varepsilon \leq \frac{1-cos\theta_{mn}}{cos\alpha_{mn}}$.

Since the linear transform to convert input data into an angle signal has a range of $[\pi/6, \pi/3]$means that $cos\theta_{mn}\ \epsilon\ [1/2\ , \sqrt{3}/2]$and $cos\alpha_{mn}\ \epsilon\ [1/2\ , \sqrt{3}/2]$; therefore [2][3][22]

$$\varepsilon \leq \frac{1-cos\theta_{mn}}{cos\alpha_{mn}} \leq \frac{1-\sqrt{3}/2}{\sqrt{3}/2} = \frac{2\sqrt{3}}{3} - 1 \approx .154 \qquad (4.3.1)$$

### 4.3.2 Extraction and testing

```
imshow(watermarkE1) % MATLAB CODE, embedding intensity e = .001
imshow(watermarkE2) % MATLAB CODE, embedding intensity e = .01
imshow(watermarkE3) % MATLAB CODE, embedding intensity e = .1
```


**Fig. 20.** Recovered images with different intensities.

In this research, we use the following methods to express the quality of our steganography [3] schema...

**Mean square error (MSE):** One method is MSE or Mean square error. The MSE measures the quality of an estimator. MSE is derived as the average of the squares of the errors per pixel, in this case, an RGB color barcode.

$$MSE = \frac{1}{MN}\sum_{iS1}^{m} \sum_{j=1}^{n} (I_{ij} - K_{ij})^2, \quad (3.4.2)$$

such that $I_{ij}$represent the matrix representation of the original image and $K_{ij}$represents the representation of the stegnafied color matrix.

**Peak Signal-to-Noise ratio (PSNR):** Another method to test the quality of the steganography scheme for images is PSNR (Peak Signal-to-Noise Ratio). The PSNR of two signals is an expression for the ratio between the greatest moment intensity of a signal and the largest moment intensity of distorting noise that affects the quality of its representation. The PSNR is expressed in terms of the decibel scale, which is logarithmic. For example, the mathematical expression for PSNR of an RGB color barcode (read as 3 stacked 2-d matrices) is:

$$PSNR = 10\, log_{10}\left(\frac{peakVal^2}{\sqrt{MSE}}\right) \quad (3.4.3)$$

**Relative similarity (RS):** The final method to test the quality of the steganography scheme for images is RS (relative similarity of two images). For images $I_1$,$I_2$is defined by:

$$RS(I_1, I_2) = 1 - \frac{\|I_2 - I_1\|_1}{\|I_1\|_1} \quad (3.4.4)$$

**Table 2:** Embedding Intensity of .001.

| **EMBEDDING $\varepsilon = .001$ (RAN 100 TIMES)** | **Original image versus embedded image** | **Watermark verus recovered watermark** | **Original image verus recovered image** |
|---|---|---|---|
| Avg Peak Signal-to-Noise Ratio (PSNR) | 40.3771 | peak signal-to-noise ratio inf | 0.8137 |
| Avg Mean Squared Error (MSE) | 3.1379 | 0 | 0.8291 |
| Average Relative Similarity (RS) | 1.0007 | .9994 | -2.0021 |

**Table 3:** Embedding Intensity of .01.

| **EMBEDDING $\varepsilon = .01$ (RAN 100 TIMES)** | **Original image versus embedded image** | **Watermark verus recovered watermark** | **Original image verus recovered image** |
|---|---|---|---|
| Avg Peak Signal-to-Noise Ratio (PSNR) | 20.3211 | peak signal-to-noise ratio inf | 0.8025 |
| Avg Mean Squared Error (MSE) | 308.1951 | 0 | 0.8037 |
| Average Relative Similarity (RS) | 1.0071 | 1.003 | -2.1034 |

**Table 4:** Embedding Intensity of .1.

| **EMBEDDING $\varepsilon = .1$ (RAN 100 TIMES)** | **Original image versus embedded image** | **Watermark verus recovered watermark** | **Original image verus recovered image** |
|---|---|---|---|

| | | | |
|---|---|---|---|
| Avg Peak Signal-to-Noise Ratio (PSNR) | -0.3531 | peak signal-to-noise ratio inf | 0.8374 |
| Avg Mean Squared Error (MSE) | 1.4554e+04 | 0 | 0.8191 |
| Average Relative Similarity (RS) | 1.0763 | 1.001 | -1.987 |

### 4.4 Image Encryption Implementation

One of the quintessen tial features of all blockchain applications is the usage of digital signature technology. A digital signature is a mathematical technique used to validate a user's authenticity and integrity in a distributed network to send a message, document, software, transaction, etc. These mathematical functions, such as the widely used RSA scheme, often rely on the difficulty of factoring large numbers but the ease of multiplying numbers. However, because of the use of quantum algorithms, namely Shor's algorithm, one could use quantum mechanics to generate a polynomial-time quantum computer algorithm for integer factorization [11].

We propose a post-quantum digital signature technology to verify the authenticity of users in a network by creating an optical encryption scheme for image security under a double random phase encoding mechanism coupled with enhanced complexity and immunity in the Fresnel domain [34].

We use the following plain-color QR (encoded with 'Sonok') and the results of the double random phase encoding, in the Fresnel domain, [23] to support our proposed method. In order to decrypt the image, the mechanism requires the encrypted image itself, the random phase mask, the Fresnel propagation parameters, and the amplitude truncation and phase truncation of the image.


**Fig. 21.** Encoded QR code.

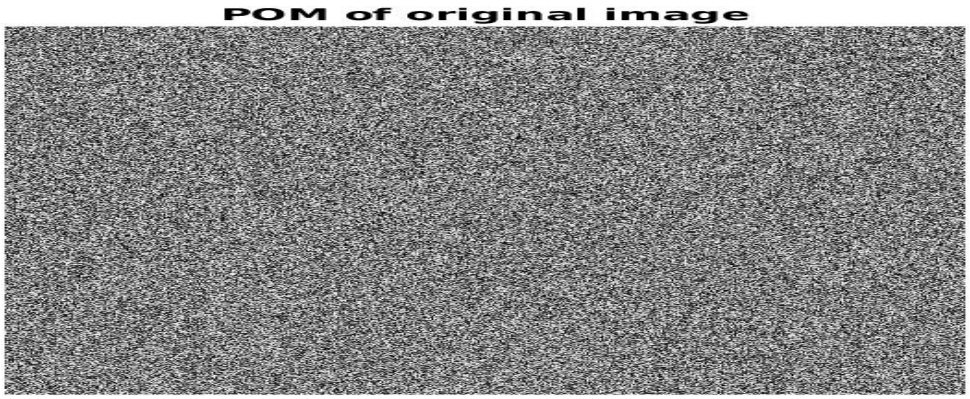


**Fig. 22.** Phase only mask of QR code.

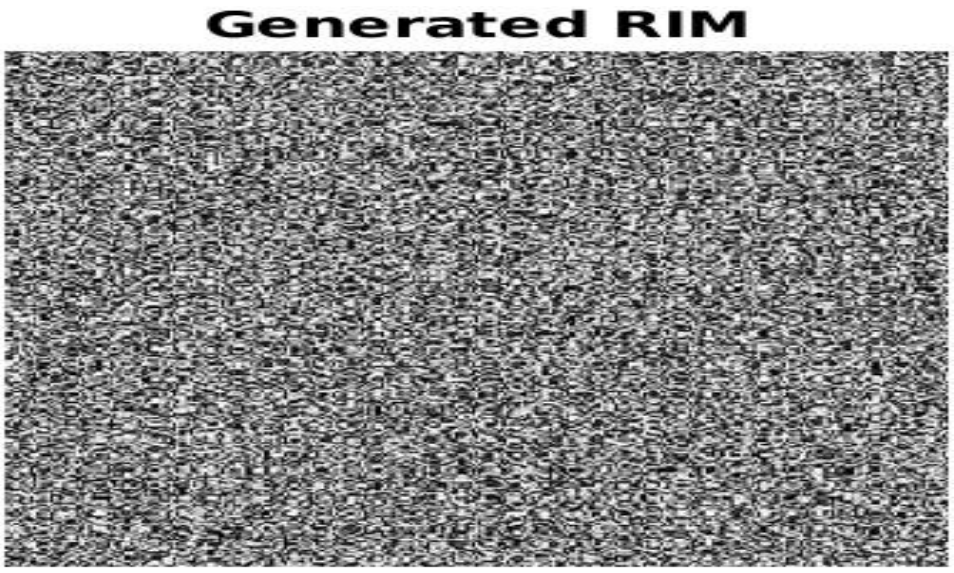


**Fig. 23.** Randomly generated intensity image.

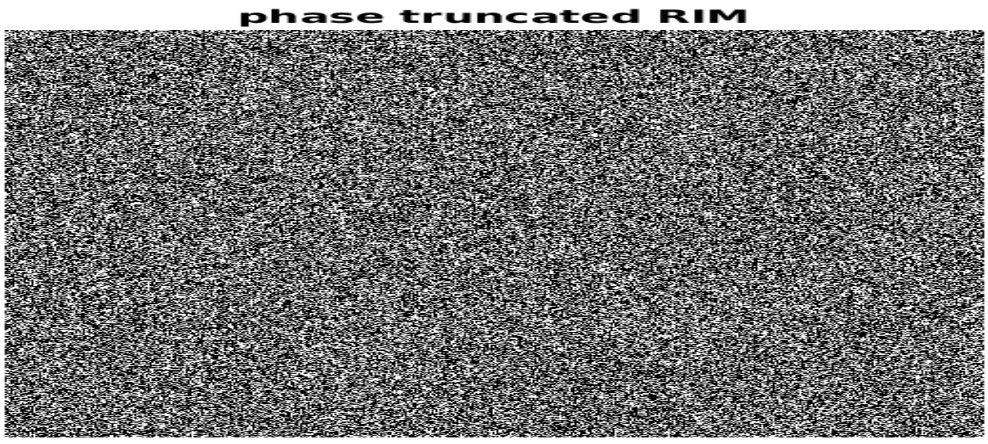


**Fig. 24.** Phase of RIM.

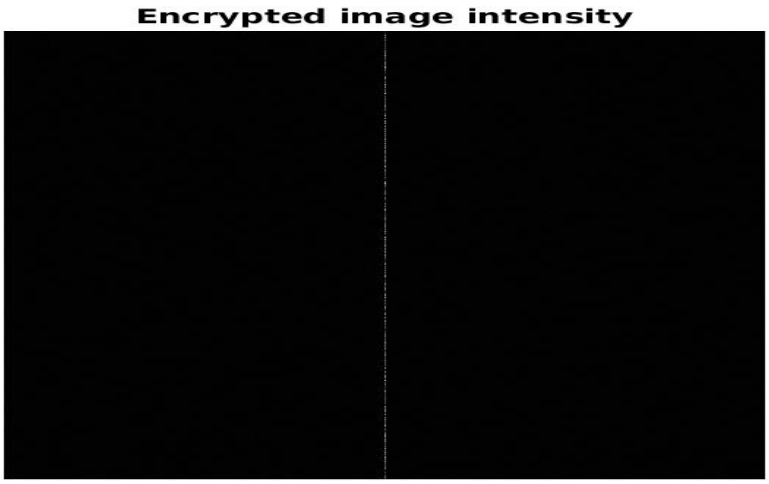


**Fig. 25**. Amplitude of image.


**Fig. 26.** Decrypted RIM.

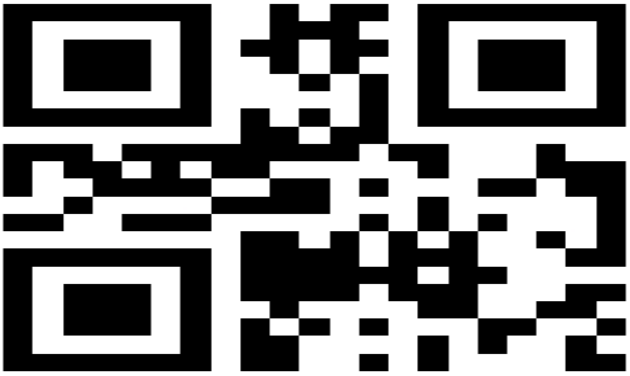

**Fig. 27.** The final recovered image.

This image, which was converted into the POM using the modified G-S phase retrieval algorithm in the FrT domain, the G-S phase retrieval algorithm, used Fresnel propagation with wavelength λ= 632 nm, propagation distance z = 50 mm. During the encryption process, the same wavelength was used as the G-S algorithm but with the free-space propagation

distances being 50 and 55 mm, respectively. Due to the nature of the Fresnel transform, the $MSE < 10^{-7}$ for all recovered images and were most likely due to errors in computing. Together with the use of image steganography, the applications in digital signature technology and encryption are apparent. Color images (including color QR codes) can be implemented in a similar fashion using separate color channels.

## 5. Conclusions and Future Research

In this article, we developed three security schemes to develop a quantum computing-resistant blockchain application. The first security scheme we developed was the use of classical to quantum mappings for zero-knowledge proof of data. This first proposed security scheme was done in accordance with finding a solution to Grover's algorithm's effect on hashing. In this research, we were able to mathematically and quantum mechanically prove the post-quantum nature of the qubit representations of classical data. We were also able to explore methods of data encryption and network storage using quantum mechanical properties. Classically we used a pseudo-quantum representation to illustrate a peer-to-peer network that's able to send, receive, encode, and decode pseudo-quantum signals of classical data. The second security scheme proposed was an optical encryption scheme for information security under the basic double random phase encoding framework using enhanced complexity and immunity in the Fresnel domain. This was done to tackle the problems created by Shor's algorithm and digital signature technology. In this research we mathematically and programmatically show the encryption of an image using a phase retrieval algorithm, double fresnel transforms, and a phase-only mask along with the reverse encryption with low error rates. The final mechanism that we used was used with the intent to enhance the security and information-carrying capability of the optical encryption scheme. In this research, we proposed a wavelet-based steganography scheme for increased storage, concealment, and limited accessibility. We used a wavelet domain to transmit pseudo-quantum signals in RGB color QRs for robust and secure data encryption capabilities. In the future, more research and testing can go into the creation and optimization of classical to quantum channels for possible applications such as in improving the hash network system found in nearly all blockchain applications. Additionally, a major goal is to create a robust implementable blockchain application for usages such as crypto exchange, housing information, user data, medical information, voting data, and the numerous and diverse number of applications that implement blockchain. Finally, our team would hope to test, optimize, and chaos test our results using both classical decryption algorithms and quantum decryption algorithms.

## 6. Acknowledgment

We would like to acknowledge the contributions of each member of the team. Sonok, a Junior at Westhill high school, was the lead author who kept direction for the entire paper. He created and coded all the privacy-preserving mechanisms and led all the experimentation in MATLAB and Python. Sonok wrote the entirety of all the sections in the paper (excluding Quantum Mechanism for zero-knowledge proof, which he worked on with Tyler Wooldridge). Sonok ran many trials, collected their results, and contributed substantial time and effort to the research.

We would like to extend our special thanks to our advisor, Dr. Wang, who has provided us with class time to learn about wavelets and machine learning, innumerable hours of assistance, and many resources. Only with Dr. Wang's aid and the Western Connecticut State University summer courses and their computer laboratory facilities could we complete the research. We are also very grateful to Avi Ray, a Junior at Westhill high school, who helped us with the generation of the

wavelet transform, and for our mentors Tyler Wooldridge and Hieu Nguyen. Tyler, a 4th-year grad student at Western Connecticut State University, worked with Sonok to write the classical to quantum mapping section, delving into its mathematical steps. Tyler also helped Sonok in the understanding of nearly a dozen papers. Hieu helped keep direction between the post-quantum mechanisms and the coding; he provided his time to review and error check code. As a result, we have learned many aspects of mathematics, quantum systems, computer science, and teamwork throughout the summer.

## 7. Appendix

Open access link to all mechanisms coded in MATLAB and python environment: https://docs.google.com/document/d/1JLcV9an9S95YLjAFHGz6IDRB-Y-hfOqBNXC1jY5QcG8/edit?usp=sharing

## REFERENCES

1. Wright CS. Bitcoin: A Peer-to-Peer electronic cash system. SSRN Electronic J. 2008. [Online]. Available: https://doi:10.2139/ssrn.3440802
2. Kenneth C. Differentially private M-Band wavelet-based mechanisms in machine learning environments. Aug. 2019.
3. Liu T, and Qiu Z. Quantum watermarking in M-band Wavelet Domain. Dongrun-Yau Science Awards (Mathematics), 2013.
4. Vadhan SP. The Complexity of differential privacy. Tutorials on the foundations of cryptography. doi:10.1007/978-3-319-57048-8_7.
5. Mallat SG. A wavelet tour of signal processing. Academic Press, 1999.
6. Rodenburg B and Stephen PP. Blockchain and quantum computing. MITRE; 2017.
7. Lin T, Xu S, Shi Q, et al. An algebraic construction of orthonormal M-band wavelets with perfect reconstruction. Appl Math Comput. 2006;172:717-730.
8. Allende López M, López D, Cerón S, et al. Quantum-resistance in blockchain networks. 2021. Online. Available: https://doi.org/10.18235/0003313
9. Sun X, Sopek M, Wang Q, et al. Towards quantum-secured permissioned blockchain: Signature, consensus, and logic. Entropy. 2019;21(9):887. Online. Available: https://doi.org/10.3390/e21090887
10. Grover LK. A fast quantum mechanical algorithm for database search. In Proc Twenty-Eighth Annual ACM Symp Theory Comput. 1996, 212-219p.
11. Shor PW. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J Comput. 1999;26(5):1484–1509.
12. Chen L, Jordan S, Liu YK, et al. Report on post-quantum cryptography. National Institute of Standards and Technology. doi:10.6028/NIST.IR.8105.
13. Bernstein DJ, Buchmann J, and Dahmen E, eds. Post-quantum cryptography. Berlin Heidelberg, Germany: Springer, 2009, doi:10.1007/978-3-540-88702-7.
14. Kirchhoff G. Zur Theorie der Lichtstrahlen. Annalen der Physik. 1883;254:663-695.

15. Brickell EF. A fastmodular multiplication algorithm with application to two key cryptography. In Proc Advan Cryptol. 1983;51–60p. [Online]. Available: https://doi.org/10.1007/978-1-4757-0602-4_5
16. Diffie W and Hellman M. New directions in cryptography (1976). *Ideas That Created the Future*. 2021;421–440p. [Online]. Available: https://doi.org/10.7551/mitpress/12274.003.0044
17. Fernandez-Carames TM, and Fraga-Lamas P. Towards post-quantum blockchain: A review on blockchain cryptography resistant to quantum computing attacks. IEEE Access. 2020;8:21091–21116. [Online]. Available: https://doi.org/10.1109/access.2020.2968985
18. Grover LK. From schrödinger's equation to the quantum search algorithm. Pramana. 2001;56(2-3):333–348. [Online]. Available: https://doi.org/10.1007/s12043-001-0128-3
19. Hwang HE. Optical color image encryption based on the wavelength multiplexing using cascaded phase-only masks in Fresnel transform domain. Optics Commun. 2012;285(5):567–573. [Online]. Available: https://doi.org/10.1016/j.optcom.2011.11.007
20. Hwang HE and Han P. Fast algorithm of phase masks for image encryption in the Fresnel domain. J Optic Soc Amer A. 2006;23(8):1870. [Online]. Available: https://doi.org/10.1364/josaa.23.001870
21. Rivest R, Shamir A, and Adelman L. A method for Obtaining digital signatures and public-key Cryptosystems (1978). Ideas That Created the Future. 2021;463–474p. [Online]. Available: https://doi.org/10.7551/mitpress/12274.003.0047
22. Yu J and Zhao S. Color QR code with pseudo quantum steganography and M-band wavelet and patch group prior based denoising. 2016. [Online]. Available: http://archive.ymsc.tsinghua.edu.cn/pacm_download/232/8531-002.pdf
23. Marco Q and Giuseppe FI. Color classifiers for 2D color barcodes. 2013 Federated Conf Comput Sci Inform Syst. IEEE Xplore, 2013.
24. Faugère JC and Joux A. Algebraic cryptanalysis of hidden field equation (HFE) cryptosystems using Gröbner bases. Advances Cryptol. 2003; 44-60p.
25. Wang X and Zhao D. Multiple-image encryption based on nonlinear amplitude-truncation and phase-truncation in fourier domain. Optics Commun. 2011;284(1):148–152. [Online]. Available: https://doi.org/10.1016/j.optcom.2010.09.034
26. Zhang P, Wang L, Wang W, et al. A blockchain system based on quantum-resistant digital signature. Secur Commun Netw. 2021;2021:1–13. [Online]. Available: https://doi.org/10.1155/2021/6671648
27. Mertz L and Young NO. Fresnel transformations of images.
28. Samuel LR. Geometrical and physical optics. Orient BlackSwan. 1986; 651p.
29. Brickell EF. A fast modular multiplication algorithm with application to two key cryptography. Advan Cryptol. 1983;51–60p. [Online]. Available: https://doi.org/10.1007/978-1-4757-0602-4_5
30. Diffie W and Hellman M. New directions in cryptography (1976). Ideas That Created the Future. 2021;421–440p. [Online]. Available: https://doi.org/10.7551/mitpress/12274.003.0044
31. Fernandez-Carames TM and Fraga-Lamas P. Towards post-quantum blockchain: A review on blockchain cryptography resistant to quantum computing attacks. IEEE Access. 2020;8:21091–21116. [Online]. Available: https://doi.org/10.1109/access.2020.2968985
32. On the quantum mechanics of collision processes.

33. Grover LK. From schrödinger's equation to the quantum search algorithm. Pramana. 2001;56(2-3):333–348. [Online]. Available: https://doi.org/10.1007/s12043-001-0128-3
34. Hwang HE. Optical color image encryption based on the wavelength multiplexing using cascaded phase-only masks in Fresnel transform Domain. Opti Commun. 2012;285(5):567–573. [Online]. Available: https://doi.org/10.1016/j.optcom.2011.11.007
35. Hwang HE and Han P. Fast algorithm of phase masks for image encryption in the Fresnel domain. J Optical Soc Amer A. 2006;23(8):1870. [Online]. Available: https://doi.org/10.1364/josaa.23.001870
36. Meltz G and Maloney WT. Optical correlation of fresnel images. Appl Opti. 1968;*7*(10):2091. [Online]. Available: https://doi.org/10.1364/ao.7.002091
37. Rivest R, Shamir A, and Adelman L. A method for obtaining digital signatures and public-key CRYPTOSYSTEMS (1978). Ideas That Created the Future, 2021;463–474p. [Online]. Available: https://doi.org/10.7551/mitpress/12274.003.0047
38. The 57 biggest data breaches in history (updated for 2021): Upguard. RSS. (n.d.). 2021. [Online]. Available: https://www.upguard.com/blog/biggest-data-breaches
39. Wang X and Zhao D. Multiple-image encryption based on nonlinear amplitude-truncation and phase-truncation in fourier domain. Opti Commun. 2011;284(1):148–152. [Online]. Available: https://doi.org/10.1016/j.optcom.2010.09.034
40. Zhang P, Wang L, and Wang W, et al. A blockchain system based on quantum-resistant digital signature. Secur Commun Netw. 2021, 1–13. https://doi.org/10.1155/2021/6671648
41. Rosistem Barcode - Barcode Education. Retrieved Sept. 20, 2016. [Online]. Available: http://www.barcode.ro/tutorials/barcodes/history.htm
42. Nielsen MA and Isaac LC. Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge: Cambridge Univ Press, 2010, doi:10.1017/cbo9780511976667.
43. Cortese JA and Braje TM. Loading classical data into a quantum computer. 2018, arXiv:1803.01958v1.
44. Mark MW. Quantum information theory. Cambridge Univ Press. 2017, arXiv:1106.1445.
45. Christian W, Stefano P, Raúl GP, et al. Gaussian quantum information. Rev Mod Phys. 2012;84(2):621–669.
46. Hwang HE, Chang HT, and Lie WN. Multiple-image encryption and multiplexing using a modified Gerchberg–Saxton algorithm and phase modulation in Fresnel-transform domain. Opt Lett. 2009;34:3917–3919.
47. Daley S. 35 blockchain applications and real-world use Cases disrupting the status quo. Retrieved Sept. 16, 2021. [Online]. Available: https://builtin.com/blockchain/blockchain-applications
48. Xue M and Zhu C. The socket programming and software design for communication based on client/server. In Pacific-Asia Conf Circuits, Commun and Syst. 2009. [Online]. Available: https://doi.org/10.1109/paccs.2009.89
49. Eiichiro F. RSA-OAEP is secure under the RSA assumption. Advances in Cryptology — CRYPTO 2001. 2001, pp. 260–274p. [Online]. Available: https://doi.org/10.1007/3-540-44647-8_16
50. Jintai D and Yang BY. Multivariate public key cryptography. Post-Quantum Cryptogr. Springer, Berlin, Heidelberg. 2009;193–241p. [Online]. Available: https://doi.org/10.1007/978-3-540-88702-7_6
51. William LH. Model-driven design & synthesis of the SHA-256 cryptographic Hash function in rewire. In Proc 27th Int Symposium Rapid Syst Prototyping: Shortening Path Specification Prototype. 2016. [Online]. Available: https://doi.org/10.1145/2990299.2990318

52. Benjamin S. Quantum coding. Phys Rev A. 1995;51(4):2738–2747. [Online]. Available: https://doi.org/10.1103/physreva.51.2738
53. Chen NA. Faithful qubit transmission in a quantum communication network with heterogeneous channels. Quantum Inf Process. 2018;17(4):2018. [Online]. Available: https://doi.org/10.1007/s11128-018-1843-8
54. Block chain. Bitcoin. Retrieved Sept 14, 2021. [Online]. Available: [Online]. Available: https://developer.bitcoin.org/devguide/block_chain.html
55. Words. Bitcoin timestamp Security. Crypto Words now WORDS. Retrieved Sept 14, 2021. [Online]. Available: https://cryptowords.github.io/bitcoin-timestamp-security
56. Security services using blockchains: A state of the art survey. IEEE Commun Surv Tutor. 2019;21(1):858 – 880. [Online]. Available: https://ieeexplore.ieee.org/document/8428402
57. Wikimedia Foundation. Grover's algorithm. 2021. Wikipedia. Retrieved Sept 14, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Grover%27s_algorithm#/media/File:Grover's_algorithm_circuit.svg
58. Wikimedia Foundation. Shor's algorithm. Wikipedia. Retrieved Sept 14, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Shor%27s_algorithm#/media/File:Shor's_algorithm.svg
59. Wikimedia Foundation. Discrete wavelet transform. Retrieved Sept 14, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Discrete_wavelet_transform#/media/File:Wavelets_-_Filter_Bank.png
60. Stock forecasting USING M-band Wavelet-Based Svr and RNN. Wavelet Transform. Retrieved Sept 14, 2021. [Online]. Available: https://arxiv.org/pdf/1904.08459
61. Mertz L and Young NO. Fresnel Transformations of Images Hologram Reconstructions. [Online]. Available: https://people.csail.mit.edu/bkph/courses/papers/Coded_Aperture/Fresnel_Transform_Mertz_Young.pdf
62. Region identification and decoding of security. Retrieved Sept 14, 2021. [Online]. Available: https://www.researchgate.net/publication/301788314_Region_Identification_and_Decoding_Of_Security_Markers_Using_Image_Processing_Tools
63. Hark D. RGBCube b.svg [Image]. 2006. [Online]. Available: https://commons.wikimedia.org/wiki/File:RGBCube_b.svg
64. Wikimedia Foundation. File: Bloch sphere.svg. Wikipedia. Retrieved Sept 14, 2021. [Online]. Available: https://en.wikipedia.org/wiki/File:Bloch_sphere.svg
65. Asymmetric encryption: Definition, architecture, usage. Okta. Retrieved Sept 14, 2021. [Online]. Available: https://www.okta.com/identity-101/asymmetric-encryption/
66. Cortese JA and Braje TM. Loading Classical Data into a Quantum Computer. 2018, arXiv:1803.01958v1.
67. Wikipedia. [Online]. Available:https://en.wikipedia.org/wiki/Quantum_logic_gate#/media/File:Quantum_Logic_Gates.png
68. Rahimyar AH, Nguyen HQ, and Wang X. Stock forecasting using M-band wavelet-based SVR and RNN-lstms models. 2nd Int Conf Inform Syst Comput Aided Edu (ICISCAE). 2019. [Online]. Available: https://doi.org/10.1109/iciscae48440.2019.221625


**Sonok Mahapatra** is currently studying with Westhill Highschool in Stamford, Connecticut. He is interested in high-level mathematics, computer science, and physics. He has published papers in multiple journals and enjoys research.

**Tyler Wooldridge** is currently a 4th-year graduate student pursuing his graduate degree in mathematics. He is interested in research in numerous fields as well as an experienced mentor for high school and college students in STEM. He hopes to pursue a Ph.D. and one day hold a position as a university professor in mathematics.

**Xiaodi Wang** received the Ph.D. degree at Michigan State University. He is currently a tenured professor in mathematics at Western Connecticut State University. He is specalizes in numerical ordinary and partial differential equations, harmonic analysis, problem solving, mathematical finance.

**Citation:** Mahapatra S, Wooldridge T, and Wang X. A Post-quantum blockchain application in m-band wavelet and Fresnel domain: A steganography-based, decentralized, distributed ledger system. Trans Eng Comput Sci. 2022;3(1):126.