Efficient Quantum Image Encryption Technique for Securing Multimedia Applications

Rakesh Saini, Bikash K. Behera, Hussein Abulkasim, Prayag Tiwari and Ahmed Farouk

Abstract—Multimedia security is a vital sector due to its significant impact on the development of industry 5.0. The current multimedia security systems depend on complex mathematical calculations, proven theoretically and practically in their inability to provide complete protection of information against internal and external attacks and penetration attempts. Unfortunately, the advancement of the quantum computer allowing the decryption of secured multimedia data by classical cryptographic algorithms will influence smart industries since no one can imagine the actual processing power of the quantum computer. Therefore, we have developed an efficient quantum image encryption technique for multimedia applications using generalized affine transform and logistic map. The designed generalized quantum circuits for the developed approach are tested and evaluated using the IBM cloud-based quantum computer. The proposed algorithms' performance and computational complexity analysis are measured and proved its efficiency against various criteria. Furthermore, a hybrid approach to reduce the circuit complexity and quantum cost using the Espresso algorithm to approximately 50% for the cost of adding one more qubit to the circuit is presented. Finally, the robustness and security analysis against various noise attacks proved that the proposed quantum image encryption method forms a secured and accurately measurable quantum image processing system.

Index Terms—Novel Enhanced Quantum Representation (NEQR), Generalized Affine Transform, Logistic Map,Quantum Cost, Complexity Analysis, Noisy Channels, Fidelity.

1 Introduction

The field of digital image processing is grown since the 1960s and has represented an important branch of information science [1]. It plays an essential role in various fields like - remote sensing, robot vision, pattern recognition, medical field, and many other areas. It is the summation of analyzing, transmitting, and manipulating digital images. Digital images are a two-dimensional array or matrix of numbers obtained by performing sampling and quantization of analog images. The exponential growth in the networked multimedia system has increased the demand for the quality and security of digital images. The quality of digital images depends on the number of pixels, and security is concerned about preventing eavesdroppers from accessing the image information. These tasks require an impenetrable cryptography technique, higher computation, and storage capacity. However, the existing computing models are limited in all these criteria. Therefore, to solve such problems, Feynman in 1982 introduced a different computing model based on quantum mechanical laws, mainly quantum superposition and quantum entanglement [2]. The main advantage of using a quantum computing model in information science is that it provides massive parallel computation [3], and unconditional security [4] to the data. That is why the quantum computing model is vastly used in information science.

Researchers developed many different ways to compute, and store the digital image in a quantum computer, such as qubit lattice [5], real ket [6], entangle image [7], flexible representation of quantum images (FRQI) [8], multi-channel representation for images (MCQI) [9], novel enhanced quantum representation (NEQR) [10], novel quantum representation of color digital images (NCQI) [11], A Generalized Model of NEQR (GNEQR) [12], normal arbitrary quantum superposition state (NAQSS) [13], and quantum Boolean image processing (QuBoIP) [14], and many more [15], [16]. Furthermore, quantum encryption protocols are also developed that transform the original image into an insignificant image to secure the image information. These encryption procedures include scrambling the pixel positions and/or changing the pixel value using the chaos theory or any other quantum transformations. There are mainly two types of encryption techniques. First, via transforming the image into frequency domain with random operations [17], [18], [19] and second, via transforming the image into a cipher image using chaos theory [19], [20], [21], [22]. Here, we are focused on the NEQR method, which is very similar to the FRQI, except in encoding the pixel values. NEQR uses eight qubits, hence 2^8 basis states to encode 256 different pixel values, while FRQI uses different polar and azimuthal angles of a single qubit state. Therefore, it is obvious that NEQR technique shows high measurement accuracy compared to FRQI.

- *Rakesh Saini was with the Department of physics, Indian Institute of Technology, (ISM), Dhanbad 826004, Jharkhand, India. E-mail: Rakesh.18ms0072@ap.iitism.ac.in*
- *Bikash K. Behera is with Bikash's Quantum (OPC) Pvt. Ltd., Balindi, Mohanpur 741246, West Bengal, India. E-mail: bikash@bikashsquantum.com*
- *Hussein Abulkasim is with the Faculty of Science, The New Valley University, El-Kharja 72511, Egypt. E-mail: hussein@scinv.au.edu.eg*
- *Prayag Tiwari is with the Department of Computer Science, Aalto University, Finland. E-mail: prayag.tiwari@aalto.fi*
- *Ahmed Farouk is with Department of Physics and Computer Science, Faculty of Science, Wilfrid Laurier University, Waterloo, Canada. E-mail: afarouk@wlu.ca*

In this paper we are working on realization of Novel Enhanced Quantum Representation (NEQR) (encoding procedure) [10], Generalized affine transform (scrambles the pixel position) [19], and logistic map (changes pixel value using chaos theory) [19]. Together the generalized affine transform and logistic map form a novel quantum encryption procedure. We presented the quantum circuits and their corresponding algorithms separately to execute these techniques on quantum computers. Then, we use the Espresso algorithm [23] used in [10] to optimize circuit complexity with our modified approach to optimize it further. Moreover, the analysis of key space, quality of encryption procedure, histogram, circuit complexity, and variation in the fidelity of the NEQR circuit under different effectiveness of the noisy channels named amplitude-damping, phase-damping, bit-flip, phase-flip, bit-phase-flip, and depolarizing is performed.

The contributions of this article are listed as follows:

1) An efficient quantum image encryption technique for multimedia applications using generalized affine transform and logistic map is proposed.
2) A hybrid approach to reduce the circuit complexity and quantum cost using the Espresso algorithm to approximately 50% for the cost of adding one more qubit to the circuit is presented.
3) The designed generalized quantum circuits for the developed approach are tested and evaluated using the IBM cloud-based quantum computer.
4) The proposed algorithms' performance, robustness, security and computational complexity analysis are measured for its efficiency against various criteria and noise attacks.

This paper is organized as follows: Section 2 briefly explains some topics that we use in this paper. Section 3 contains the implementation of NEQR, encryption, and decryption using generalized affine transform and logistic map on a quantum computer. The analysis of the key space, quality of encryption procedure, histogram, circuit complexity, and robustness test of NEQR circuit in different noisy channels are presented in Section 4. Finally, we conclude in Section 5 and discuss future works.

2 Preliminaries

2.1 Novel Enhanced Quantum Representation (NEQR)

The NEQR is a quantum gray image representation model that offers a methodology [10], which uses a gray image (size: $2^n \times 2^n$) parameters as input and encode them to a quantum system or circuit that uses superposition and entanglement to represent the image parameters. The circuit is build of $2n + 8$ qubits. 8-qubits are used to store the gray scale value $f(Y, X)$ of the pixels, and 2n-qubits are for the pixel position (Y, X). The final state of this quantum circuit, which represents the image, is given as:

$$|\psi\rangle = \frac{1}{2^n} \sum_{Y,X=0}^{2^{2n}-1} |f(Y,X)\rangle |YX\rangle \tag{1}$$

where, $|f(Y,X)\rangle$ can be written in the bit sequence as, $\bigotimes_{i=0}^{p-1} |C^i_{YX}\rangle = |c^7_{YX} c^6_{YX} \dots c^1_{YX} c^0_{YX}\rangle$, which represents the encoded gray scale value of pixel, and

$$\begin{aligned} |YX\rangle &= |Y\rangle |X\rangle \\ &= |Y_{n-1} Y_{n-2} \dots Y_1 Y_0\rangle |X_{n-1} X_{n-2} \dots X_1 X_0\rangle \end{aligned}$$

is to store the corresponding vertical and horizontal coordinate values of the pixels.

Algorithm 1: Algorithm to encode digital image into a quantum circuit using NEQR.

Input : Classical image: $2^n \times 2^n$
Declare the Pixel value: $f(Y, X)$
Declare the vertical and horizontal location of pixels: (Y, X)
Quantum Circuit: QC $\{$initial state $|0\rangle^{2n+8}\}$
Output: Quantum Circuit for given classical image

1 Start by transforming the position and pixel value in binary sequence:
(Y, X) $\rightarrow |YX\rangle$ and $f(Y, X) \rightarrow \bigotimes_{i=0}^{7} |C^i_{YX}\rangle$
2 To store this binary numbers in the quantum circuit follow
(I) The initial state of the circuit: $|0\rangle^{2n+8}$
(II) Apply Hadamard gate on 2n-qubits that generate superposition of 2^{2n} states:
$|0\rangle^8 \otimes (H^{2n} \otimes |0\rangle^{2n}) \rightarrow \frac{1}{2^n} \sum_{Y=0}^{2^n-1} \sum_{X=0}^{2^n-1} |0\rangle^8 \otimes |YX\rangle$
(III) Define an operation O_{YX} that apply a unitary operation U_{YX} on the pixel value qubits for pixel state $|YX\rangle$ as:
$O_{YX} = I \otimes \sum_{j=0}^{2^{2n}-1} \sum_{i=0, ij \neq YX}^{2^{2n}-1} |ij\rangle \langle ij| + U_{YX} \otimes |YX\rangle \langle YX|$

$U_{YX} |0\rangle \leftarrow |0 \oplus C_{YX}\rangle$
(IV) Now, operate O_{YX} on all qubits:
$\frac{1}{2^n} O_{YX} \left(\sum_{Y=0}^{2^n-1} \sum_{X=0}^{2^n-1} |0\rangle^8 \otimes |YX\rangle \right) \rightarrow$
$\frac{1}{2^n} \sum_{Y=0}^{2^n-1} \sum_{X=0}^{2^n-1} (U_{YX} \otimes |0\rangle^8) \otimes |YX\rangle \rightarrow$
$\frac{1}{2^n} \sum_{Y=0}^{2^n-1} \sum_{X=0}^{2^n-1} |f(Y,X)\rangle \otimes |YX\rangle \rightarrow |\psi\rangle$
3 **return** QC, $|\psi\rangle$

As an example, we have taken an image of size 2×2 as shown in Fig. 6a, and after performing the NEQR, it can be represented by the quantum state:

$$\begin{aligned} |\psi_I\rangle = \frac{1}{2}(&|11111111\rangle |00\rangle + |00000000\rangle |01\rangle \\ &+ |11001000\rangle |10\rangle + |01100100\rangle |11\rangle) \end{aligned} \tag{2}$$

The Algo. {1} explains the NEQR procedure more briefly and logically by clearly presenting the encoding of digital images into a quantum circuit (see Fig. 1) step by step.

2.2 Generalized Affine Transform

The generalized affine transform shown in Algo. {2} scrambles the pixel locations $|YX\rangle$ of an image. Let us say X and Y represent the actual horizontal and vertical location of pixels, respectively. Then we can write the mathematical

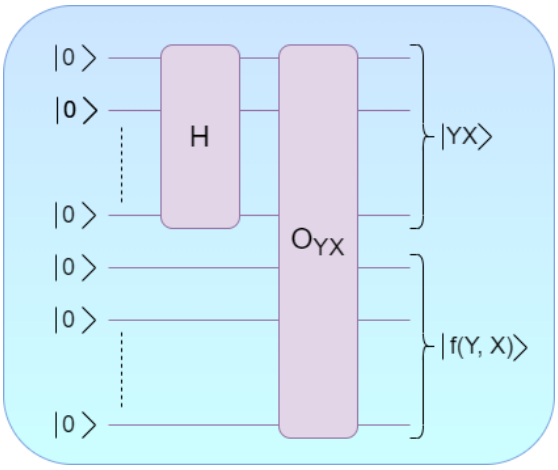


Fig. 1: NEQR procedure circuit using $(2n + 8)$-qubits. Here, H is the Hadamard gate and O_{YX} is a controlled gate that operate U_{YX} on pixel qubits, when 2n-qubits are in the state $|YX\rangle$, to obtain $f(Y, X)$.

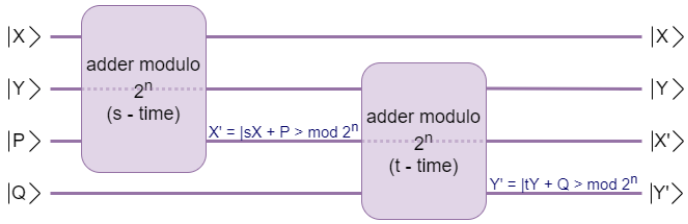


Fig. 2: The figure shows a General quantum circuit for generalized affine transform to scrambled the original pixel coordinate $|YX\rangle$ into $|Y'X'\rangle$. The dotted qubit line in the circuit represents bypassing the operation.

representation of a generalized affine transform according to [19] as:

$$\begin{pmatrix} X' \\ Y' \end{pmatrix} = \begin{pmatrix} s & 0 \\ 0 & t \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} + \begin{pmatrix} P \\ Q \end{pmatrix} \pmod{2^n} \tag{3}$$

Here, X' and Y' represent the scrambled horizontal and vertical locations of pixels and take the form (sX + P) and (tY + Q), respectively, with modulo addition 2^n. The s, t, P & Q are parameters of the generalized affine transformation, which obey the following conditions:

- P and Q should not be zero. In the circuit, to store the P & Q value, we use n-qubits for each.
- s and t should be co-prime with 2^n.

The generalised circuit in Fig. 2 shows the operations to scramble the original coordinates $|X\rangle, |Y\rangle$ to $|X'\rangle, |Y'\rangle$ as calculated in Eq. (3). To recover the original pixel coordinates, the inverse of generalised affine transform shown in Algo. {4} is performed. Which follow the following mathematical terminology:

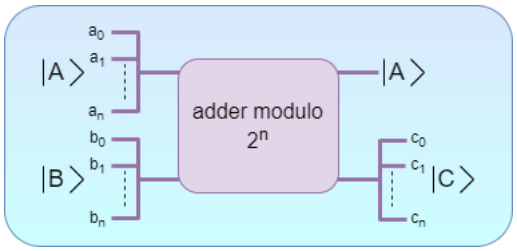
$$\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} s^{-1} & 0 \\ 0 & t^{-1} \end{pmatrix} \begin{pmatrix} X' - P \\ Y' - Q \end{pmatrix} \pmod{2^n} \tag{4}$$

This equation gives the original coordinates of the pixel X = s^{-1}(X' - P) mod 2^n and Y = t^{-1}(Y' - Q) mod 2^n. Here, s^{-1} and t^{-1} are modular multiplicative inverse [24] of s and t, respectively and can be calculated as:

$$\begin{aligned} 1 &= ss^{-1} \mod 2^n \\ 1 &= tt^{-1} \mod 2^n \end{aligned} \tag{5}$$

The generalised circuit in Fig. 4 shows the performed operations to recover the original coordinates $|X\rangle, |Y\rangle$ from $|X'\rangle, |Y'\rangle$ as calculated in Eq. (4).

(a)



(b)

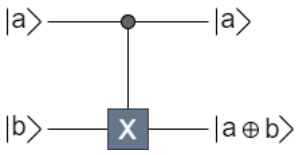


Fig. 3: (a) Adder modulo circuit: with output $|C\rangle = |A\rangle \oplus |B\rangle$ (mod 2^n). (b) Quantum CNOT gate: gives output as (a $\oplus$ b) mod 2.

Algorithm 2: Algorithm to encrypt the pixel position by using genralized affine transform.

Input : Declare the size of the image: $2^n \times 2^n$
Declare the generalized affine transform parameters: P,
Q, s and t
Declare the horizontal and vertical location of pixels: X, Y

Output: Image of scrambled pixel location

initialization
$i, j \leftarrow 1$
while $i \leq s$ **do**
  $X' = (iX + P) \bmod 2^n$
  $i \leftarrow i + 1$
end
while $j \leq t$ **do**
  $Y' = (jY + Q) \bmod 2^n$
  $j \leftarrow j + 1$
end
return X', Y'

2.3 Adder modulo 2^n

Adder modulo 2^n for two binary numbers with n-bits each defines such that the resultant added binary number has no change in the number of bits. This can be done by ignoring the last carry bit. A generalized quantum circuit for adder modulo 2^n is given in [25]. A general black-box representation of it for the input of 2n-bits is shown in Fig. 3a and for n = 1, the adder modulo 2^n circuit is equivalent to the quantum CNOT gate shown in Fig. 3b.

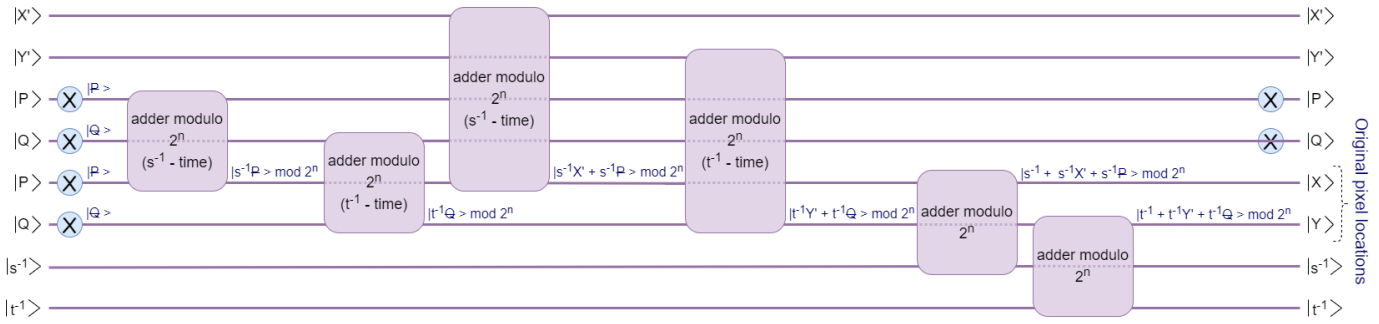


Fig. 4: The quantum circuit represents a General decryption procedure for generalized affine transform to obtain the real pixel coordinate $|YX\rangle$ from $|Y'X'\rangle$. The doted qubit line refers to avoiding the applied operation.

2.4 Logistic Map

It is used to map the value of previous step to the next step [22] using the following dynamical equation:

$$L_{\eta+1} = \delta L_\eta (1 - L_\eta) \tag{6}$$

where $\eta = 0, 1, 2, ..., 2^{2n} - 1$, $L_0 \epsilon \{0, 1\}$ is taken as the initial value of the logistic map, δ is the growth rate and 2^{2n} is the total number of pixels. This dynamical equation behaves chaotically for $3.85 \leq \delta \leq 4$, which means that with this growth rate the dynamical Eq. (6) generates pseudo-random strings of 0's and 1's.

3 Implementation on Qunatum Computer

3.1 NEQR Circuit

Following the image in Fig. 6a that we have used first and this image has n = 1. So, we take a quantum circuit of $2 + 8$-qubits initialized in the state with all 0's, whose first 2-qubits store the pixel coordinates and the rest are for the pixel values. The detailed procedure of encoding the image parameters in a quantum circuit following from Algo. {1} is:

1) The image has 4-pixels with different positions and values. To accomplish all of the pixel positions, four different states are required. So, we apply the Hadamard gate on the first 2-qubits that generate a superposition quantum system of four equally probable states ($|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$).
2) Now, Toffoli gates are applied with coordinate qubits as control and the rest qubits as the target. These Toffoli gates transformed the pixel value qubits from state $\bigotimes_{i=0}^{7} |0^i\rangle$ to the state $\bigotimes_{i=0}^{7} |C_{YX}^i\rangle$, corresponding to each pixel ($|YX\rangle$).

The conversion of the state of the quantum circuit is as follows:

$$\begin{aligned} |0000000000\rangle &\xrightarrow[\text{first 2 - qubits}]{\text{apply H-gate on}} |00000000\rangle\,|{+}{+}\rangle \\ &\xrightarrow[\text{on rest qubits}]{\text{apply multi-control gates}} |\psi_I\rangle \end{aligned} \tag{7}$$

The resultant quantum circuit for this whole procedure is shown in Fig. 8a. This circuit has 14-Toffoli gates, so before executing on real superconducting devices, it first should transpile to the basis one and two qubit gates of that device. However, after passing through the transpiler, the circuit size and depth increase exponentially, which affects execution efficiency and fidelity enormously.

To overcome this problem, we use the same image compression procedure used in [10]. This procedure deals with the control bit strings of the pixel values by using the Espresso algorithm [23] such that a similar task can be performed with lower control bit string. As a result, the circuit complexity is reduced by using the control string given by the Espresso algorithm.

In this compression procedure, we have added an extra step to lower the circuit complexity further. This step uses an ancillary qubit to store the Toffoli gates information. We then use this ancillary qubit as control and targeting the desired qubits one by one using C-NOT gates. We can perform the same task that performs multiple Toffoli gates in the circuit by using few Toffoli gates, C-NOT gates and one ancillary qubit, as an example see Fig. 8c. This step is used when the number of Toffoli gates with the same control bit string are still high after using the image compression procedure [10]. Fig. 8b and Fig. 8d show an comparison between image compression and after adding our step in this compression.

3.2 Encryption procedure

The encryption procedure uses generalized affine transform and logistic map to transform the original image into a cipher image. Here we discuss and present algorithms for encryption schemes and how they affect image parameters.

Algorithm 3: Algorithm to encrypt pixel values by using logistic map.

Input : Declare the size of the image: $2^n \times 2^n$
Declare the logistic map parameters: δ, L_0
Output: J_{YX}, T_{YX} $\{YX = \eta\}$

initialization
$J, T = [\ \ ]$
foreach η *in range*$(2^{2n}$*-1)* **do**
  $L_{\eta+1} = \delta L_\eta (1 - L_\eta)$
  $J_{YX} = round(mod(L_\eta \times 2^8, 2^8))$
  $J.append(J_{YX})$
end
foreach η *in range*$(2^{2n}$*-1)* **do**
  $T_{YX} = J_{2^{2n}-1-YX}$
  $T.append(T_{YX})$
end
return J, T

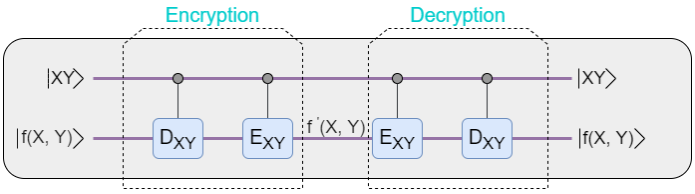


Fig. 5: Logistic map for encryption and decryption of pixel value guided by horizontal (X) and vertical (Y) position ofpixels.

3.2.1 logistic map (Diffusion stage)

The diffusion stage works on the pixel value and control by the corresponding pixel coordinates. This stage works on chaos theory as described in Section 2.4, so the resultant image becomes chaotic. The required quantities to operate logistic map in a quantum circuit is calculated according to Algo. {3}. The operations of the logistic map in a quantum circuit are as follows:

1) Pick the output lists J, T of Algo. {3}.
2) Now, transform each J_{YX} and T_{YX} form J and T to binary sequence as $|J_{YX}\rangle = \otimes_{i=0}^{p-1} |J^i_{YX}\rangle, |T_{YX}\rangle = \otimes_{i=0}^{p-1} |T^i_{YX}\rangle$.
3) Define two quantum operation D_{YX} and E_{YX} as

$$\begin{aligned} D_{YX}\,|C^i_{YX}\rangle &\rightarrow |C^i_{YX} \oplus J^i_{YX}\rangle \quad \text{, and} \\ E_{YX}\,|C^i_{YX} \oplus J^i_{YX}\rangle &\rightarrow |C^i_{YX} \oplus J^i_{YX} \oplus T^i_{YX}\rangle \end{aligned} \tag{8}$$

4) Define functions that operate D_{YX} and E_{YX},

$$\begin{aligned} \phi^1_{YX} &= I \otimes \frac{1}{2^n} \sum_{j=0}^{2^n-1} \sum_{i=0, ji \neq YX}^{2^n-1} |ji\rangle\langle ji| + D_{YX} \\ &\otimes |YX\rangle\langle YX| \\ \phi^2_{YX} &= I \otimes \frac{1}{2^n} \sum_{j=0}^{2^n-1} \sum_{i=0, ji \neq YX}^{2^n-1} |ji\rangle\langle ji| + E_{YX} \\ &\otimes |YX\rangle\langle YX| \end{aligned}$$

5) Now, operate ϕ^1_{YX} and ϕ^2_{YX} one by one on the quantum state ψ_I as

$$\begin{aligned} |\psi_L\rangle &= \phi^2_{YX}\phi^1_{YX}\,|\psi_I\rangle \\ &= \frac{1}{2^n} \sum_{Y,X=0}^{2^{2n}-1} (E_{YX}D_{YX} \bigotimes_{i=0}^{p-1} |C^i_{YX}\rangle)\,|YX\rangle \\ &= \frac{1}{2^n} \sum_{Y,X=0}^{2^{2n}-1} \bigotimes_{i=0}^{p-1} |C^i_{YX} \oplus J^i_{YX} \oplus T^i_{YX}\rangle\,|YX\rangle \end{aligned}$$

3.2.2 Generalized Affine Transform (Permutation stage)

let A represents the operator for generalized affine transform. As we already know that generalized affine transform only affects the pixel positions of the image. When we are operating A on the quantum state of the image $|\psi_L\rangle$, it only focuses on the position of the pixels $|YX\rangle$. The operation is as follow:

$$\begin{aligned} |\psi_{AL}\rangle = A\,|\psi_L\rangle &= \frac{1}{2^n} \sum_{Y=0}^{2^n-1} \sum_{X=0}^{2^n-1} |f'(Y,X)\rangle\, A\,|YX\rangle \\ &= \frac{1}{2^n} \sum_{Y=0}^{2^n-1} \sum_{X=0}^{2^n-1} |f'(Y,X)\rangle\,|Y'X'\rangle \end{aligned} \tag{9}$$

where, A scrambles the pixels horizontal and vertical positions as $A\,|X\rangle = |X'\rangle$ and $A\,|Y\rangle = |Y'\rangle$, respectively. This transformation is explained in Algo. {2} and the output of this algorithm $|X'\rangle$ and $|Y'\rangle$ are given by:

$$\begin{aligned} |X'\rangle &= |sX + P\rangle \mod 2^n \\ |Y'\rangle &= |tY + Q\rangle \mod 2^n \end{aligned}$$

According to the quantum version of the generalized affine transform to scramble pixel positions, a general quantum circuit given in Fig. 2, is designed. This quantum circuit performs adder modulo 2^n s and t times to transform the states $|X\rangle \rightarrow |X'\rangle$ and $|Y\rangle \rightarrow |Y'\rangle$, respectively, as shown in Fig. 2. The algebra of this quantum circuit is as follows:

$$\begin{aligned} |X, P\rangle &\xrightarrow[\text{s - times}]{\text{adder modulo } 2^n} |X, (sX + P) \mod 2^n\rangle \\ |Y, Q\rangle &\xrightarrow[\text{t - times}]{\text{adder modulo } 2^n} |Y, (tY + Q) \mod 2^n\rangle \end{aligned}$$

3.3 Decryption procedure

In this section, the inverse generalized affine transform and inverse logistic map procedure are given. This procedure helps the selected parties to decrypt the encrypted image. Here, we first describe the inverse generalized affine transform then the inverse logistic map procedure.

3.3.1 Inverse Generalized Affine Transform

The inverse generalized affine transform outputs the image's original horizontal and vertical pixel coordinates, as we have discussed before. let us say A^{-1} shows inverse generalized affine transform operation.

$$\begin{aligned} |\psi_L\rangle &= A^{-1}\,|\psi_{AL}\rangle \\ &= \frac{1}{2^n} \sum_{Y,X=0}^{2^{2n}-1} |f'(Y,X)\rangle\, A^{-1}\,|Y'X'\rangle \\ &= \frac{1}{2^n} \sum_{Y,X=0}^{2^{2n}-1} |f'(Y,X)\rangle\,|YX\rangle \end{aligned} \tag{10}$$

According to theorem stated in [17] the negative adder modulo 2^n appeared in the Eq. (4) can be realized as follows:

$$(x - y) \mod 2^n = (x + (\overline{y} + 1)) \mod 2^n \tag{11}$$

where, $\overline{y} = \overline{y}_{n-1}\overline{y}_{n-2}\ldots\overline{y}_0 = 1 - y_i$ with i = n-1, n-2,. . . ,0. Then, X = s^{-1}(X' - P) mod 2^n is converted to X = s^{-1}(X' + ($\overline{P}$ + 1)) and similarly Y. The circuit shown in Fig. 4 take $2s^{-1}+3$ steps to obtain the original pixel position $|X\rangle$ and $2t^{-1}+3$ for $|Y\rangle$. The Algo. {4} presents the step by step explanation of inverse generalized affine transform procedure.

Algorithm 4: Algorithm to decrypt the pixel values using generalized affine transform.

Input : Declare the generalized affine transform parameters: P, Q, s and t
Declare the scrambled image pixel location: X', Y'
Quantum Circuit for location of pixels: QC_1 (QC_1 has 2n qubits)
Quantum Circuit to store P and Q: QC_2 (QC_2 has 2n qubits, n to store P and rest for Q)
Generalized affine transform operator: A (Adder modulo 2^n)
Output: Original image

Apply A on $\overline{P}$ {as control} to $\overline{P}$ {as target}
foreach *i in range*$(1, 2s^{-1}+4)$ **do**
  foreach $1 \leq i \leq s^{-1}$ **do**
    $\overline{P} \leftarrow (i\overline{P}) \bmod 2^n$
  end
  for $i = s^{-1}+1$ **do**
    Replace $\overline{P}\{control\}$ to $X'\{control\}$
  end
  foreach $s^{-1}+2 \leq i \leq 2s^{-1}+1$ **do**
    $\overline{P} \leftarrow (iX' + s^{-1}\overline{P}) \bmod 2^n$
  end
  for $i = 2s^{-1}+2$ **do**
    Replace $X'\{control\}$ to $s^{-1}\{control\}$
  end
  for $i = 2s^{-1}+3$ **do**
    $\overline{P} \leftarrow (s^{-1} + s^{-1}X' + s^{-1}\overline{P}) \bmod 2^n$
  end
end
do $\overline{P} \leftarrow \overline{Q}$, $s^{-1} \leftarrow t^{-1}$, $X' \leftarrow Y'$ repeat from 1 to 18 **for** Y
return $\overline{P}$ = X and $\overline{Q}$ = Y

3.3.2 *logistic map*

The inverse logistic map procedure is very simple with the output of Algo. {3}. This procedure operates the same functions ϕ^1_{YX} and ϕ^2_{YX} used in the encryption procedure (see Fig. 5) as:

$$\begin{aligned}|\psi\rangle &= \phi^1_{YX}\phi^2_{YX}\,|\psi_L\rangle \\ &= \phi^1_{YX}\phi^2_{YX}\phi^2_{YX}\phi^1_{YX}\,|\psi\rangle \\ &= \frac{1}{2^n}\sum_{Y,X=0}^{2^{2n}-1}(D_{YX}E_{YX}E_{YX}D_{YX}\bigotimes_{i=0}^{p-1}|C^i_{YX}\rangle)\,|YX\rangle \\ &= \frac{1}{2^n}\sum_{Y,X=0}^{2^{2n}-1}\bigotimes_{i=0}^{p-1}|C^i_{YX}\rangle\,|YX\rangle \end{aligned} \tag{12}$$

4 Analysis

4.1 Correlation Coefficient Analysis

The correlation coefficient (r) of two image is the analysis of how much this images are correlated to each other. The mathematical formula for correlation coefficient is

$$r = \frac{\sum(x_i - x_m)(y_i - y_m)}{\sqrt{\sum(x_i - x_m)^2 \sum(y_i - y_m)^2}} \tag{13}$$

Here, x_i is the i_{th} pixel intensity in plain/original image, y_i is the i_{th} pixel intensity in cipher image, x_m & y_m are the mean pixels intensity of the plain and cipher image.

pixel position $\lvert YX\rangle$	plain image	cipher image
$\lvert 00\rangle$	255	213
$\lvert 01\rangle$	0	37
$\lvert 10\rangle$	200	237
$\lvert 11\rangle$	100	78
mean value	139	141

TABLE I: Table shows the pixel intensity of plain and cipher image corresponding to there position in the image.

One can decide the correlation based on the value of r such that, if r = 1: Its mean that both images are identical to each other, or if r = 0: Its mean that both are completely uncorrelated, or if r = -1: Its mean that both images are negative of each other. After using the values of each pixel intensity form the Table I, the value of correlation coefficient (r) is -1, which means that the cipher image in Fig. 6c is negative image of the original image in Fig. 6a.

4.2 Analysis of Quality of Encryption Procedure

The quality of the encryption procedure depends on the size of key space, key sensitivity, and ability to generate different cipher images for a slight change in the plain image. We have calculated the following quantities to analyze it:

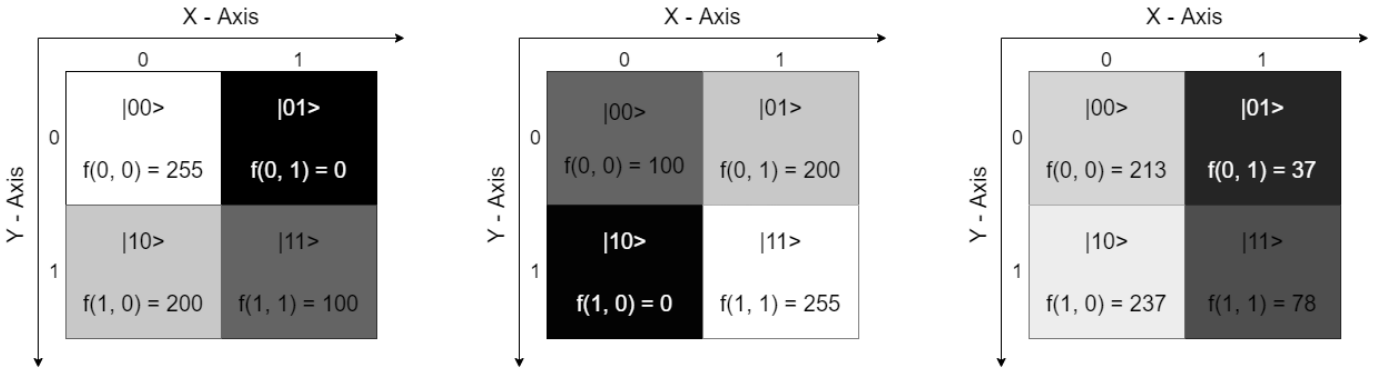
4.2.1 *Key Space*

key space can be understood as the storage space that consist all the possible combinations of keys for an encrypted system such that one of them can decrypt the system perfectly. Like - If someone encrypts data using a unique n-bit string of 0's and 1's, then to access the plain data without any knowledge of the key, someone has to try all possible combinations of n-bits, in total 2^n combinations or brute-force attack in real-time. As the number of bits (n) rises consecutively, the key space also rises. So, the key space can be written as:

$$keyspace(KS) \propto 2^n \tag{14}$$

The analysis of key space for our case is as follows,

- For generalized affine transform we uses n_P, n_Q, n_s, and n_t qubits to store the parameters P, Q, s, and t respectively. So, the key space for generalized affine transform is $KS_A = 2^{n_P+n_Q+n_s+n_t}$.
- For logistic map we uses n_{L_0} and n_δ qubits for L_0 and δ, respectively. So, here the key space is $KS_L = 2^{n_{L_0}+n_\delta}$.
- In total the key space is $KS_T = KS_A + KS_L$
- For a good encryption of the image, one should select the encryption parameters (P, Q, s, t, L_0, δ) such that, the brute-force attack that work by trying all possible combinations of the key, can not break the encryption in the real-time.
- However, as discussed in [19] that L_0 and δ are infinite decimals, so the key space becomes infinite. Resultantly the cipher image is secured to the brute-force attack and attacks of its kind.



(a) (b) (c)

Fig. 6: The figure shows three different images (a) Original image for Eq. (2). (b) encrypted image using the generalized affine transformation. (c) Encrypted image using both generalized affine transform and logistic map. Here, we have performed generalized affine encryption with P, Q, s, t = 1 and logistic map encryption with L_0 = 0.5557924316949603 and δ = 3.9816188727791215.

4.2.2 *Differential Analysis*

The differential analysis to check the sensitivity of encryption procedure for encryption key and plain image is performed with the help of two experiments. In the first experiment, the change is made in the pixel value of the pixel at position (0, 0) from 255 to 254 and the resultant cipher image is shown in the Fig. 7a. Fig. 7b shows the differential image between image of original cipher image 6c and the cipher image after change in pixel value 7a. The difference can be measured with the help of number of pixels change rate (NPCR) and unified average changing intensity (UACI), and these quantities are calculated by,

$$NPCR = \frac{1}{W \times H}\left[\sum_{i,j} D(i,j)\right] \times 100\%$$
$$UACI = \frac{1}{W \times H}\left[\frac{\sum_{i,j} |C(i,j) - C'(i,j)|}{255}\right] \times 100\% \quad (15)$$

where, W and H represents width and height of the image, respectively. D(i, j) is determine as,

$$D(i,j) = \begin{cases} 1 & \text{for} \quad C(i,j) = C'(i,j) \\ 0 & \text{for} \quad C(i,j) \neq C'(i,j) \end{cases} \quad (16)$$

$C(i,j)$ and $C'(i,j)$ are for the pixel value at the coordinate (i, j) of image in Fig. 6c and in Fig. 7a.

The second experiment for the encryption key sensitivity is to change the key L_0 from 0.5557924316949603 to 0.6, while the other keys are intact. The corresponding encrypted image is shown in Fig. 6c and calculated the difference with original cipher image. According to the differential image shown in Fig. 7d, we can say that the encryption procedure generate completely different image if one of the keys is changed.

The NPCRs and UACIs are listed in Table II, which demonstrates the considered novel encryption method is not much sensitive to the plain text. However it is highly sensitive to the encryption keys.

image	NPCR (in %)	UACI (in %)
cipher image (same parameter)	25	0.098
cipher image (different parameter)	100	28.04

TABLE II: Table shows the values of NPCR and UACI. In first raw, for the images in Fig. 6c and in Fig. 7a. In second raw, for the images in Fig. 6c and in Fig. 7c.

4.3 Analysis of Histogram

The analysis of histogram includes the reconstruction of the image using the outcome information from the histogram. The order of the state of the histogram is as $|q_9 \dots q_2 q_1 q_0\rangle \rightarrow |C_0 \dots C_7 Y X\rangle$ and by converting this state from binary to decimal, the image parameter pixel location (YX) and value (f(Y, X)) can be re-calculated.

4.4 Complexity analysis

To measure the complexity of the quantum circuit, the two following quantities are defined:

1) Quantum Cost: Number of basis operations used in the quantum circuit.
2) Time Complexity: Depth or number of time steps in the execution of the quantum circuit. Execution of basis operations in one step is taken as 1-unit.

4.4.1 *Quantum cost*

In our case, the part of the circuit, which affects the quantum cost, is the encoding pixel values as per the pixel positions. Because this part includes several $\mathbb{C}^{2n}$ - NOT gates. These gate decomposed into 2(2n - 1) Toffoli gates and a c-NOT gate with 2n - 1 ancillary qubit, in which $\mathbb{C}^{2n}$ represents the 2n control qubits. So, in total, the Decomposition of $\mathbb{C}^{2n}$ - NOT gate into basis single and two-qubit gates will increases the cost enormously.

This increase in cost can be optimized with the help of the Espresso algorithm. The algorithm takes the control qubits information and gives an output, which performs the same control task but with fewer control qubits. For example, we have shown the NEQR circuit in Fig. 8a, which has

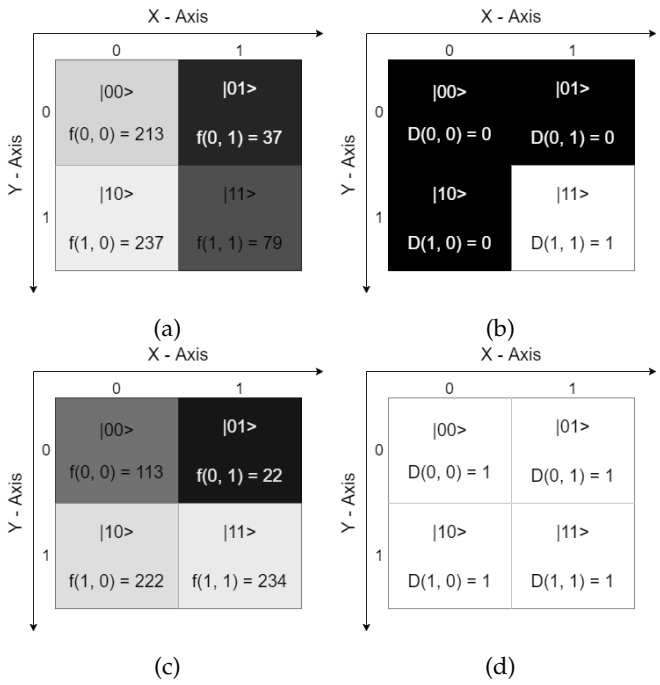


Fig. 7: The figure shows two cipher and two differential images: (a) is the cipher image of Fig. 6a with change in pixel value of pixel at position (0, 0) from 255 to 254. (b) is the differential image of Fig. 7a and Fig. 6c. (c) is the cipher image of Fig. 6a with change in key L_0 from 0.5557924316949603 to 0.6, while the other keys are intact. (d) is a differential image of Fig. 7c and Fig. 6c.

14 - Toffoli gates, and by the Espresso algorithm, the same circuit is optimized to Fig. 8b, which has 8 - Toffoli gates. In quantity, without the Espresso algorithm, the quantum cost for quantum circuit shown in Fig. 8a is 195. After the application of the Espresso algorithm, it decreases to 134. furthermore, to optimize quantum cost more, we have to use the technique shown in Fig. 8c and using this method, our circuit is converted to as shown in Fig. 8d with quantum cost 59. This decrease in quantum cost is considerable because, with the Espresso algorithm and the presented technique, the overall cost of NEQR quantum circuit shown in Fig. 8a is decreased to 30.26%. Which is 68.72% without adding technique shown in Fig. 8c.

4.4.2 *Time complexity:*

As discussed in [10] the time complexity of NEQR circuit is no more than $\mathcal{O}(qn2^{2n})$ and in [19] that the time complexity of encryption procedure is $\mathcal{O}(n)$ and same for the decryption procedure.

The quantitative analysis of time complexity for the NEQR circuit, after encryption and after decryption, is presented in Table III. Also, after including technique shown in Fig. 8c, the total complexity is optimized enormously with the cost of adding one ancillary qubit. From Table III, we can see that the time complexity is decreased to 22.19%. Which is 43.71% without adding technique shown in Fig. 8c.

	circuit	Quantum cost	Time complexity
Normal	NEQR	195	142
	Encryption	681	511
	Decryption	1173	883
with Espresso	NEQR	134	91
	Encryption	342	238
	Decryption	554	386
including presented technique	NEQR	59	37
	Encryption	176	116
	Decryption	298	196

TABLE III: Table shows the computational Complexity, in first raw without any optimization, in second raw with Espresso algorithm and in third with the Espresso algorithm & with the presented method showed in Fig. 8c.

4.5 Robustness test against different noise

The actual computation on a real quantum computer, acquire noise. So, In this section, we are going to execute our circuit in six different types of noisy environments named amplitude-damping, phase-damping, bit-flip, phase-flip, bit-phase-flip, and depolarizing. These noises can be characterized by Kraus operators [26]. Therefore to see the effect of particular noise on a circuit, the corresponding Kraus operator should affect each qubit of the circuit or the qubits in which we desired to see the impact of noise. This noise introduced in the system environment converts the pure quantum state ($|\psi\rangle$) to a mixed state, whose properties are best describe by the density matrix ($\rho = |\psi\rangle\langle\psi|$) representation. The effect of the noise on the pure state ψ can be stated as:

$$\epsilon^r(\rho) = \sum_m (\otimes_{i=0}^n K_m^{rq_i}) \rho (\otimes_{i=0}^n K_m^{rq_i\dagger}) \tag{17}$$

where,K_m represents the Kraus operators for the noise, r represents the type of noisy environment, q_i represents the qubit on which noise acts and n represents the number of qubits. The effect of noise on the real quantum state $|\psi\rangle$ can be quantify by calculating the fidelity. Fidelity is measure of closeness between two quantum states which is given by:

$$F(\rho, \epsilon(\rho)) = \left[Tr\sqrt{\sqrt{\rho}\epsilon(\rho)\sqrt{\rho}} \right]^2 \tag{18}$$

Here, $\rho = |\psi\rangle\langle\psi|$ is ideal state, $\epsilon(\rho)$ is the noisy state, and $F(\rho, \epsilon(\rho)) \in \{0, 1\}$, where 1 means both states are identical, and 0 means these are entirely different.

4.5.1 *Against Amplitude-Damping Environmental Noise*

Amplitude-damping noise in the channel causes energy dissipation (loss of energy) in the effected quantum system. The Kraus operators for amplitude-damping noise are:

$$K_0^A = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma_A} \end{bmatrix}; \quad K_1^A = \begin{bmatrix} 0 & \sqrt{\gamma_A} \\ 0 & 0 \end{bmatrix} \tag{19}$$

$$\epsilon^A(\rho) = \otimes_{i=0}^9 K_0^{Aq_i} \rho \otimes_{i=0}^9 K_0^{Aq_i\dagger} + \otimes_{i=0}^9 K_1^{Aq_i} \rho \otimes_{i=0}^9 K_1^{Aq_i\dagger} \tag{20}$$

Where, $\gamma_A \in (0, 1)$ is the probability of energy dissipation of the quantum system. The quantum state in Eq. (2) after passing through the noise environment is shown in the Eq. (20).

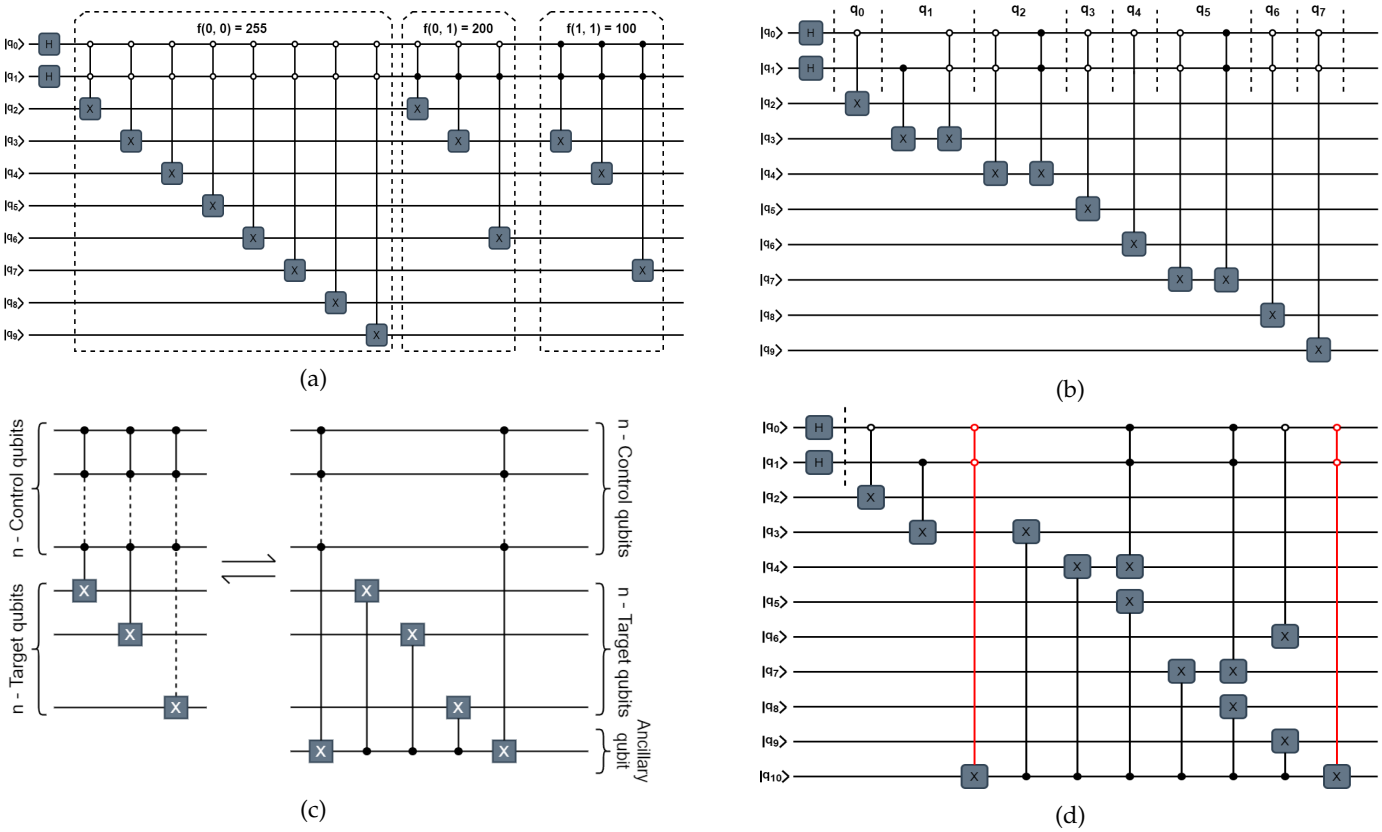


Fig. 8: (a) Quantum circuit for classical image shown in Fig. 6a, according to the NEQR quantum model. Note that the position coordinate of pixel are encoded to the circuit as $|YX\rangle \rightarrow |q_1 q_0\rangle$ and the pixel values as $\left|C^0_{YX} C^1_{YX} \dots C^7_{YX}\right\rangle \rightarrow |q^9 q^8 \dots q^2\rangle$. (b) Quantum circuit represents the circuit shown in Fig. 8a after the reduction in the controlled information for the pixel values using the Espresso algorithm. (c) Figure shows the quantum circuit to reduce the number of Toffoli gate with the help of one ancillary qubit. (d) Figure shows the quantum circuit after the application of Espresso algorithm and the presented technique in Fig.8c. The Toffoli gate represented by red is the one whose information is saved in the ancillary qubit.

Fig. 9 shows the fidelity change as the probability of energy dissipation goes from 0 to 1. It is also observed from the fidelity diagram that the fidelity decreases as the dissipation probability increases.

4.5.2 Against Phase-Damping Environmental Noise

Phase damping noisy Chanel introduces the loss of relative phase information of the quantum state and no loss in the energy of it. The Kraus operators for phase damping noise are:

$$K_0^P = \begin{bmatrix} \sqrt{1-\gamma_P} & 0 \\ 0 & \sqrt{1-\gamma_P} \end{bmatrix}; \quad K_1^P = \begin{bmatrix} \sqrt{\gamma_P} & 0 \\ 0 & 0 \end{bmatrix}$$
$$K_2^P = \begin{bmatrix} 0 & 0 \\ 0 & \sqrt{\gamma_P} \end{bmatrix} \tag{21}$$

$$\epsilon^P(\rho) = \otimes_{i=0}^{10} K_0^{Pq_i} \rho \otimes_{i=0}^{10} K_0^{Pq_i\dagger} + \otimes_{i=0}^{10} K_1^{Pq_i} \rho \otimes_{i=0}^{10} K_1^{Pq_i\dagger} \tag{22}$$

where $\gamma_P \in (0, 1)$ is the probability of loosing relative Phase information. After passing through this noise channel the quantum state is given by Eq. (22) and the fidelity F^P of the output state is calculated using Eq. (18). In the Fig. 9, we have shown the change in fidelity as the decoherence rate of the phase-damping noise changes.

4.5.3 Against Bit-Flip Environmental Noise

Bit-flip noisy channel flips the state of the quantum bit with the probability $\gamma_B \in (0, 1)$. The Kraus operators for bit-flip noise has the following form:

$$K_0^B = \begin{bmatrix} \sqrt{1-\gamma_B} & 0 \\ 0 & \sqrt{1-\gamma_B} \end{bmatrix}; \quad K_1^B = \begin{bmatrix} 0 & \sqrt{\gamma_B} \\ \sqrt{\gamma_B} & 0 \end{bmatrix} \tag{23}$$

$$\epsilon^B(\rho) = \otimes_{i=0}^{9} K_0^{Bq_i} \rho \otimes_{i=0}^{9} K_0^{Bq_i\dagger} + \otimes_{i=0}^{9} K_1^{Bq_i} \rho \otimes_{i=0}^{9} K_1^{Bq_i\dagger} \tag{24}$$

The quantum state after transmitted through the bit-flip noisy channel is written in the Eq. (24). The fidelity of the output state is can be calculated using Eq. (18) and the Fig. 9 shows the variation in the fidelity as the probability of flipping the bit γ_B varies.

4.5.4 Against Phase-Flip Environmental Noise

In the bit-flip noisy channel we have seen that the state of quantum is flipped, similar in this type of noise channel the relative phase of the quantum system flips instead of its

state. The Kraus operators for this noise is give as:

$$K_0^W = \begin{bmatrix} \sqrt{1-\gamma_W} & 0 \\ 0 & \sqrt{1-\gamma_W} \end{bmatrix}; \quad K_1^B = \begin{bmatrix} \sqrt{\gamma_W} & 0 \\ 0 & -\sqrt{\gamma_W} \end{bmatrix} \tag{25}$$

$$\epsilon^W(\rho) = \otimes_{i=0}^{9} K_0^{Wq_i} \rho \otimes_{i=0}^{9} K_0^{Wq_i\dagger} + \otimes_{i=0}^{9} K_1^{Wq_i} \rho \otimes_{i=0}^{9} K_1^{Wq_i\dagger} \tag{26}$$

The quantum state no longer a pure state after transmitting through this noisy channel, so it is more prior to written it in density form as Eq. (26). The fidelity of this output state is shown in Fig. 9 and can be calculated by following Eq. (18).

4.5.5 Against Bit-Phase-Flip Environmental Noise

This noisy channel flips both the state of the qubit and relative phase of the quantum system. The Kraus operators for this type of noisy environment are give as:

$$K_0^F = \begin{bmatrix} \sqrt{1-\gamma_F} & 0 \\ 0 & \sqrt{1-\gamma_F} \end{bmatrix}; \quad K_1^F = \begin{bmatrix} 0 & -\iota\sqrt{\gamma_W} \\ \iota\sqrt{\gamma_W} & 0 \end{bmatrix} \tag{27}$$

$$\epsilon^F(\rho) = \otimes_{i=0}^{9} K_0^{Fq_i} \rho \otimes_{i=0}^{9} K_0^{Fq_i\dagger} + \otimes_{i=0}^{9} K_1^{Fq_i} \rho \otimes_{i=0}^{9} K_1^{Fq_i\dagger} \tag{28}$$

The mixed state after passing through this noisy channel is written in Eq. (28). The variation in the fidelity as per the flipping probability $\gamma_F \in (0, 1)$ is shown in the Fig. 9, which is calculated by using Eq. (18).

4.5.6 Against Depolarization Environmental Noise

In Depolarizing noisy environment the qubits are depolarized with probability $\gamma_D \in (0, 1)$. The Kraus operators are as follow:

$$K_0^D = \begin{bmatrix} \sqrt{1-\gamma_D} & 0 \\ 0 & \sqrt{1-\gamma_D} \end{bmatrix}; \quad K_1^D = \begin{bmatrix} 0 & \sqrt{\frac{\gamma_D}{3}} \\ \sqrt{\frac{\gamma_D}{3}} & 0 \end{bmatrix}$$

$$K_2^D = \begin{bmatrix} 0 & -\iota\sqrt{\frac{\gamma_D}{3}} \\ \iota\sqrt{\frac{\gamma_D}{3}} & 0 \end{bmatrix}; \quad K_3^D = \begin{bmatrix} \sqrt{\frac{\gamma_D}{3}} & 0 \\ 0 & \sqrt{\frac{\gamma_D}{3}} \end{bmatrix} \tag{29}$$

$$\epsilon^D(\rho) = \otimes_{i=0}^{9} K_0^{Dq_i} \rho \otimes_{i=0}^{9} K_0^{Dq_i\dagger} + \otimes_{i=0}^{9} K_1^{Dq_i} \rho \otimes_{i=0}^{9} K_1^{Dq_i\dagger} \tag{30}$$

The output quantum state from this noisy channel is written in Eq. (30). The fidelity is calculated by the Eq. (18) and Fig. 9 shows the plot between fidelity and probability of depolarizing the qubits.

It can be noted that all noise simulations are only done for the NEQR circuit. We did not include the rest of the circuit in this noise simulation because of the memory and execution time limitations. We observe this problem because the dimension of the density matrix becomes $2^{15} \times 2^{15}$ and computation with this size of the matrix is require high memory space and computation power.

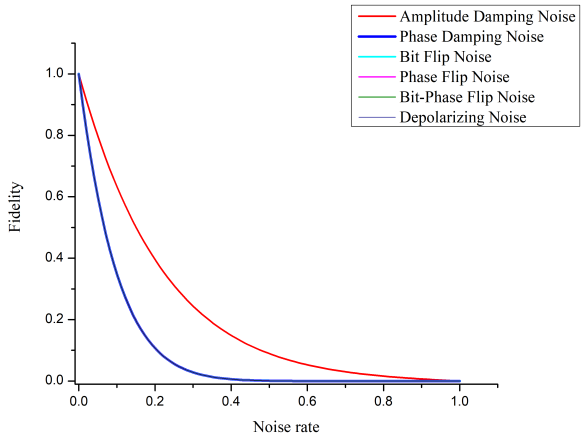


Fig. 9: The plot shows the variation in the fidelity for the quantum state shown in Eq. (2) with the change of noise rate in six different types of noisy channels. Here the fidelity variation is come out to be similar for all the noisy environment except the Amplitude damping noisy environment.

5 Conclusion

In this paper, we have implemented the novel enhanced quantum representation (NEQR) for a $2^n \times 2^n$ image, a novel quantum image encryption, and its decryption procedure on IBM's quantum computers. These protocols are part of the fittest techniques to represent and encrypt a digital image in the view of their measurement accuracy and chaotic output with high key space & sensitivity, respectively. We have shown how to realize each of these procedures on a quantum computer and provided general quantum circuits and algorithms for them. The multi-control-NOT gates that appear in the quantum circuit are essential to encode the pixel values, whose decomposition into basis one and two qubit gates increases the circuit complexity and further increases enormously as n increases. However, we have provided a method combined with the image compression method used by Zhang *et al.* to lower the effect of decomposition on circuit complexity and successfully reduced to approximately 50% as compared to Zhang *et al.* for a 2×2 test image on the cost of adding one more qubit to the circuit. However, it is very effective for circuit complexity to make this tradeoff between a little extra quantum storage and a significant reduction in circuit complexity. The analysis of encryption procedure demonstrated that the scheme is sensitive to the encryption key and has enough large key space that it can resist ordinary attacks and attacks similar to the brute-force attack. We conclude that the NEQR combined with the discussed novel quantum image encryption method form a secured and accurately measurable quantum image processing system. In the future, we would like to demonstrate the implementation of more quantum image representation protocols with an encryption procedure on the quantum computer. We believe it helps us to come up with a quantum image processing system with a lower circuit complexity, higher measurement accuracy, and susceptible encryption technique that can be realizable on a quantum computer.

References

[1] R. C. Gonzalez, and R. E. Woods, Digital Image Processing, Publishing House of Electronics Industry, Beijing (2002).
[2] R. P. Feynman, Simulating physics with computers, Int. J. Theor. Phys., **21**(6/7), 467-488 (1982).

[3] D. Deutsch, Quantum theory, the Church Turing principle and the universal quantum computer, Proc. R. Soc. Lond. A, Math. Phys. Sci., **400**, 97-1171 (1985).

[4] N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, Quantum cryptography, Reviews of modern physics, **74**(1), 145, (2002).

[5] S.E. Venegas-Andraca, and S. Bose, Storing, processing and retrieving an image using quantum mechanics, Proceeding of the SPIE Conference Quantum Information and Computation, 137-147 (2003).

[6] J. I. Latorre, Image compression and entanglement, arXiv:quant-ph/0510031 (2005).

[7] S. E. Venegas-Andraca, and J. L. Ball, Processing images in entangled quantum systems, Quantum Inf. Process., **9**(1), 1-11 (2010).

[8] P. Q. Le, F. Y. Dong, and K. Hirota, A flexible representation of quantum images for polynomial preparation, image compression and processing operations, Quantum Inf. Process., **10**(1), 63-84 (2011).

[9] B. Sun, A. M. Iliyasu, F. Yan, F. Y. Dong, and K. Hirota, An RGB multi-channel representation for images on quantum computers, J. Adv. Comput. Intell. Intell. Inf., **17**(3), 404-417 (2013).

[10] Y. Zhang, K. Lu, Y. H. Gao, and M. Wang, NEQR: a novel enhanced quantum representation of digital images, Quantum Inf. Process., **12**(12), 2833-2860 (2013).

[11] J. Sang, S. Wang, and Q. Li, A novel quantum representation of color digital images, Quantum Inf. Process., **16**(2), 42 (2017).

[12] H.-S. Li, P. Fan, H.-Y. Xia, H.-L. Peng, and S. X. Song, Quantum implementation circuits of quantum signal representation and type conversion, IEEE Trans. Circuits Syst. I, Reg. Papers, vol., **66**, no. 1, 341–354, (2019).

[13] H. S. Li, Q. X. Zhu, R. G. Zhou, S. Lan, and X. J. Yang, Multi-dimensional color image storage and retrieval for a normal arbitrary quantum superposition state, Quantum Inf. Process., **13**(4), 991-1011 (2014).

[14] M. Mastriani, Quantum Boolean image denoising, Quantum Inf. Process., **14**(5), 1647-1673 (2014).

[15] F. Yan, A. M. Iliyasu, and S. E. Venegas-Andraca, A survey of quantum image representations, Quantum Inf. Process., **15** (2016).

[16] J. Su, X. Guo, C. Liu and L. Li, A New Trend of Quantum Image Representations, IEEE Access, **8** (2020).

[17] N. Jiang, and L. Wang, Analysis and improvement of the quantum Arnold image scrambling, Quantum Inf. Process., **13**, 1545-1551 (2014).

[18] Y. G. Yang, X. Jia, S. J. Sun, and Q. X. Pan, Quantum cryptographic algorithm for color images using quantum Fourier transform and double random-phase encoding, Inf. Sci., **277**, 445-457 (2014).

[19] H.-R. Liang, X.-Y. Tao, and N.-R. Zhou, Quantum image encryption based on generalized affine transform and logistic map, Quantum Inf. Process., **15**(7), 2701-2724 (2016).

[20] Q. W. Ran, L. Wang, J. Ma, *et al.*, A quantum color image encryption scheme based on coupled hyper-chaotic Lorenz system with three impulse injections, Quantum Inf. Process., **17**, 188 (2018).

[21] N. R. Zhou, W. W. Chen, X. Y. Yan, and Y. Q. Wang, Bit-level quantum color image encryption scheme with quantum cross-exchange operation and hyper-chaotic system, Quantum Inf. Process., **17**, 137 (2018).

[22] R. L. Devaney, An Introduction to Chaotic Dynamical Systems, Westview Press, Boulder (2003).

[23] R. K. Brayton, A. Sangiovanni-Vincentelli, C. McMullen, and G. Hachtel: Log. Minimization Algorithms VLSI Synth, Kluwer Academic Publishers, Dordrecht (1984).

[24] K. H. Rosen, Elementary Number Theory and Its Applications, United Kingdom, Pearson/Addison Wesley (2005).

[25] N. Jiang, and L. Wang, Analysis and improvement of the quantum Arnold image scrambling, Quantum Inf. Process., **13**(7), 1545-1551 (2014).

[26] K. Kraus, States, Effects and Operations, Springer-Verlag, Berlin, (1983).

Rakesh Saini received his M.Sc degree in physics from the Indian Institute of Technology (ISM) Dhanbad in 2020. He started his research in quantum computing during his masters. He is currently involved in few research projects in quantum computing. His research interest includes simulating theories of quantum computation and quantum information on quantum computers, quantum machine learning, and quantum technologies.

Bikash K. Behera received the BS-MS dual degree in 2018 from Indian Institute of Science Education and Research (IISER) Kolkata, Mohanpur 741246, Nadia, West Bengal, India. Currently, he is the founder, CEO and director at Bikash's Quantum (OPC) Private Limited, Balindi, Mohanpur 741246, Nadia, West Bengal, India. His research interests include quantum simulation, quantum machine learning, quantum communication, quantum error correction, quantum cryptography, quantum chemistry, quantum blockchain, to name a few.

Hussein Abulkasim received the B.S., M.S.,and Ph.D. degrees in computer science from SouthValley University, in 2004, 2012, and 2016, respectively. He was a Lecturer with the College of Computer Sciences and Information Systems at Jazan University, from 2013 to 2014. Abulkasim worked as a Lecturer with the Department of Mathematics and Computer Science at Assiut University from 2014 to 2017. From 2017 to 2019, he was an Assistant Professor with the Department of Mathematics and Computer Science at New Valley University. He is currently a Postdoctoral Research Fellow with the Cybersecurity Research Lab at Ryerson University, Canada. His current research interests include quantum cryptography, quantum computation and communication, blockchain technology, and IoT security

Ahmed Farouk is currently assistant professor, before that he was a Postdoctoral Research Fellow at Wilfrid Laurier University and Ryerson University, Canada. He received his M.Sc. and Ph.D. degrees from Mansoura University, Egypt. He is one of the Top 20 technical co-founders of the Quantum Machine Learning Program by Creative Destruction Lab at the University of Toronto. Furthermore, he is selected as Top 25 of InnovateTO150 Canada to showcase the best of Toronto's next generation of change-makers, innovators, and entrepreneurs. He is exceptionally well known for his seminal contributions to theories of Quantum Information, Communication, and Cryptography. He published 62 papers in reputed and high impact journals like Nature Scientific Reports and Physical Review A. The exceptional quality of his research is recognized nationally and internationally. He selected by the scientific review panel of the Council for the Lindau Nobel Laureate Meetings to participate in the 70th Lindau Nobel Laureate Meeting. His volunteering work is apparent since he appointed as chair of the IEEE computer chapter for the Waterloo-Kitchener area and editorial board for many reputed journals like Nature Scientific Reports, IET Quantum Communication, and IEEE Access. Also, he selected for IEEE and IET Young Professional Ambassador and as a moderator for the new IEEE TechRxiv. Recently, he appointed as an associate editor for the IEEE Canadian Review (ICR).