

# Quantum Visual Cryptographic System for Secure Digital Image Protection

K. Sai Pranav<sup>1</sup>, P. Sasi Kumar<sup>2</sup>,

<sup>1</sup>Final year MCA Student, Department of Computer Applications, GIET Engineering College, Andhra Pradesh, India

<sup>2</sup> Assistant Professor, Department of Computer Applications, GIET Engineering College, Andhra Pradesh, India

\*\*\*

**Abstract** - The rapid growth of digital communication has increased the need for secure image transmission and storage mechanisms. Traditional encryption techniques provide security but may become vulnerable with the advancement of computational technologies. This project presents a Quantum Visual Cryptography System that integrates quantum computing concepts with image encryption to enhance data confidentiality and protection.

The proposed system generates a random cryptographic key using quantum circuits implemented through Qiskit and the Aer Simulator. The generated quantum key is further processed using the SHA-256 hashing algorithm to produce a secure key stream. During encryption, the original image is divided into cryptographic shares using visual cryptography principles and XOR-based operations. One share is generated randomly, while the second share is created using the original image, the random share, and the quantum-derived key stream. The encrypted shares are stored separately, ensuring that the original image cannot be reconstructed without both shares and the correct quantum key. For decryption, the authorized user provides the quantum key, which is expanded and combined with the encrypted shares to recover the original image. The system also includes a graphical user interface developed using Python Tkinter for user interaction and performance analysis through execution time visualization. The proposed approach demonstrates how quantum-generated randomness can be integrated with visual cryptography techniques to improve image security. The system provides confidentiality, secure key management, and efficient image reconstruction, making it suitable for secure multimedia communication and digital image protection applications.

**Key Words:** *Quantum Computing, Visual Cryptography, Image Encryption, Quantum Key Generation, Qiskit, SHA-256, XOR Encryption, Image Security, Secure Image Transmission, Multimedia Data Protection.*

## 1. INTRODUCTION

Digital images are used everywhere today, from social media and online communication to healthcare, banking, and cloud storage systems. As the use of digital images continues to increase, protecting image data from unauthorized access has become an important requirement. Sensitive images may contain personal, financial, or confidential information, and if such data falls into the wrong hands, it can lead to privacy and security issues. Therefore, developing secure methods for image protection has become a significant area of research. Many traditional encryption techniques are available for securing digital data. Although these methods provide a good level of security, researchers are constantly looking for stronger approaches to deal with evolving cyber threats. One of the emerging technologies in this field is quantum computing. Quantum computing introduces new concepts such as

superposition and quantum randomness, which can be utilized to generate highly unpredictable cryptographic keys.

The main idea of this project is to combine quantum computing concepts with visual cryptography to improve image security. Visual cryptography protects an image by dividing it into multiple shares. Individual shares do not reveal useful information about the original image, making it difficult for unauthorized users to access the actual content. Only when the required shares are combined correctly can the original image be reconstructed.

In this project, a Quantum Visual Cryptography System is developed using Python. The system uses Qiskit to generate a quantum-based random key. The generated key is further processed using the SHA-256 hashing algorithm to create a secure key stream. During the encryption process, a random share is generated and combined with the original image using XOR operations. The encrypted data is stored as separate shares, providing an additional layer of protection.

A graphical user interface has been developed using Tkinter to make the system easy to use. Users can upload images, generate quantum keys, perform encryption and decryption operations, and view the results through the interface. The project also includes execution-time analysis to evaluate the efficiency of the encryption process.

The objective of this project is to demonstrate how quantum-generated randomness and visual cryptography can work together to provide enhanced image security. The proposed system offers secure image sharing, reliable image recovery, and improved protection of sensitive image data. Such a solution can be useful in areas where confidentiality and secure image transmission are important requirements.

## 2. METHODOLOGY

The proposed Quantum Visual Cryptography System follows a sequence of steps to ensure secure image encryption and decryption. The methodology combines quantum key generation, cryptographic hashing, visual cryptography, and XOR-based encryption techniques to protect digital images from unauthorized access. The system is designed to provide a secure environment for image storage and transmission while maintaining ease of use through a graphical user interface. Each stage of the methodology plays an important role in ensuring confidentiality, integrity, and successful image reconstruction.

### 2.1. Image Acquisition

The first step in the system is image acquisition. The user selects an image through the graphical user interface developed using Python Tkinter. The uploaded image is loaded into the system using the OpenCV library and prepared for further processing. This image acts as the input for the encryption process.

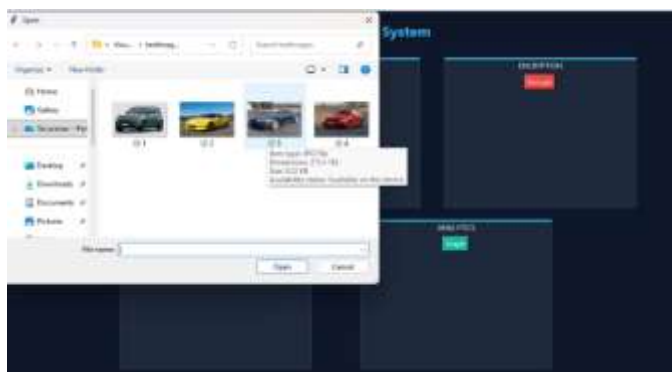


Image acquisition is an important stage because it serves as the starting point for all subsequent operations. The system accepts digital images and converts them into a matrix of pixel values that can be processed computationally. Each pixel contains color information represented in numerical form, allowing mathematical and cryptographic operations to be performed efficiently. Once the image is successfully loaded, it is displayed within the user interface to confirm successful image selection.

## 2.2. Quantum Key Generation

After uploading the image, a cryptographic key is generated using quantum computing principles. The Qiskit framework is used to create a quantum circuit consisting of multiple qubits. Hadamard gates are applied to place the qubits into a superposition state, allowing the generation of highly random measurement outcomes. The generated quantum bit sequence is converted into byte data and processed using the SHA-256 hashing algorithm. This produces a secure hexadecimal key that serves as the encryption key for the system. The generated key is also stored for later use during decryption.



The use of quantum-generated randomness increases the unpredictability of the encryption key. Since the encryption process depends heavily on the strength of the generated key, this stage contributes significantly to the overall security of the system. The generated key can also be copied and stored by the user for future decryption purposes.

## 2.3. Key Expansion

The generated quantum key is transformed into a key stream suitable for image encryption. Since image files contain a large number of pixel values, the SHA-256 output is expanded to match the size of the image data. This expanded key stream ensures that every pixel component participates in the encryption process. The key expansion process allows the encryption algorithm to operate uniformly across the entire image. By generating a key stream equal to the image size, the system ensures that all image pixels are influenced by the

cryptographic key. This improves the effectiveness of the encryption process and prevents direct exposure of image patterns.

## 2.4. Image Encryption

The encryption process begins by creating a random share with the same dimensions as the original image. This share contains randomly generated pixel values and acts as the first encrypted component. The second share is generated using XOR operations between the original image pixels, the random share, and the expanded quantum key stream. Mathematically, the encryption process can be represented as:

Encrypted Share = Original Image XOR Random Share XOR Quantum Key Stream. The resulting encrypted shares are stored separately. Since neither share contains sufficient information to reveal the original image independently, the confidentiality of the image is preserved.



The encryption operation is performed on every pixel of the image. As a result, the encrypted output appears completely random and does not reveal any visible characteristics of the original image. The use of XOR operations provides computational efficiency while maintaining the ability to accurately reconstruct the image during decryption.

## 2.5. Secure Storage of Shares

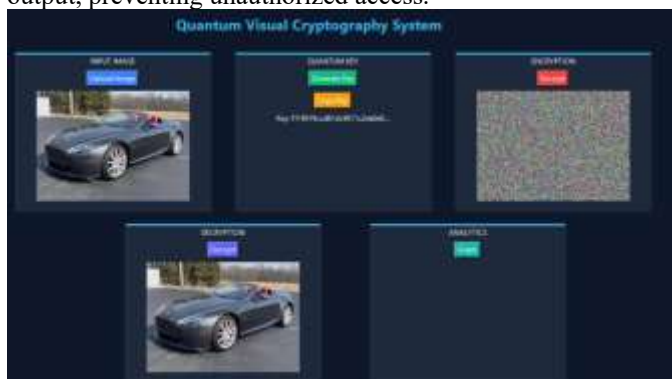
The generated shares are saved as separate files in the system. Storing the shares independently increases security because an attacker must obtain both shares and the correct encryption key to reconstruct the original image. The separation of encrypted shares introduces an additional layer of protection. Even if one share becomes accessible to an unauthorized user, meaningful image information cannot be extracted from it. This approach minimizes the risk of data leakage and strengthens the overall security of the image protection framework.

## 2.6. Image Decryption

To recover the original image, the user provides the quantum-generated key through the graphical interface. The system loads the encrypted shares and regenerates the key stream using the provided key.



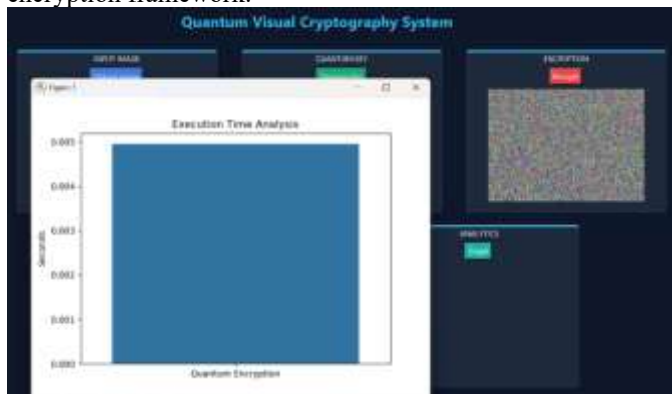
The original image is reconstructed by applying reverse XOR operations on the encrypted shares and the regenerated key stream. If the correct key is entered, the original image is successfully restored. An incorrect key produces an invalid output, preventing unauthorized access.



The decryption process demonstrates the reversibility of the XOR operation. Since the same key stream used during encryption is required for reconstruction, only authorized users possessing the correct key can successfully recover the original image. This mechanism ensures secure access control and protects sensitive image information.

## 2.7. Performance Analysis

The execution time of the encryption process is measured using Python timing functions. The collected performance data is displayed graphically using Matplotlib and Seaborn libraries. This analysis helps evaluate the efficiency of the proposed encryption framework.



Performance evaluation is important for understanding the practicality of the system. By measuring execution time, the system provides information regarding the computational cost of image encryption. The graphical representation makes it easier to observe the performance characteristics of the proposed approach and assess its suitability for real-world applications.

## 2.8. Result Visualization

The graphical user interface displays the original image, encrypted image share, and decrypted image. This allows users to visually verify the effectiveness of the encryption and decryption processes. The interface also provides options for key generation, key copying, image reconstruction, and performance analysis.

Result visualization improves user interaction and system usability. By presenting images before and after encryption, users can clearly observe the transformation of image data. The visualization also confirms successful image reconstruction during decryption, thereby validating the correctness of the implemented methodology.

## 3. DISCUSSION

The decryption process showed that the method can accurately restore the original image when the correct quantum key and encrypted shares are used. The reconstructed image kept its original visual quality, showing that the encryption and decryption steps maintained the image's integrity. Performance tests indicated that the system completed encryption and decryption tasks within a reasonable time frame. Combining quantum key generation, hashing, XOR operations, and visual cryptography offered a good balance between security and efficiency. Based on the results, the proposed system is suitable for applications that need secure image protection, such as managing confidential documents, storing medical images, securing cloud data, and ensuring safe communication systems. In conclusion, the project shows that integrating quantum-inspired techniques with visual cryptography can greatly improve image security while ensuring reliable performance and accurate image reconstruction.

## 4. CONCLUSIONS

In this project, a Quantum Visual Cryptography System was created to ensure secure encryption and decryption of images. The system uses quantum-based key generation, SHA-256 hashing, XOR encryption, and visual cryptography methods to increase the security of digital images. By combining these technologies, the approach provides multiple levels of protection against unauthorized access. The implementation successfully encrypted images, created secure cryptographic shares, and accurately restored the original image during decryption using the correct quantum key. The encrypted image and shares did not contain any meaningful data on their own, thus improving the confidentiality of the information. The experimental results show that the system achieves a good balance between security and efficiency while preserving image quality during reconstruction. The use of quantum-inspired randomness makes the encryption process more robust and greatly reduces the possibility of predicting the key. Overall, the system offers a dependable and effective method for securely storing and transmitting images. The project demonstrates the potential of merging quantum computing ideas with visual cryptography for modern cybersecurity applications. Future improvements could involve real-time image processing, cloud integration, and more advanced quantum algorithms to further boost security and performance.

## 5. REFERENCES

1. A JB, Raj C, Sukumaran R, G SM (2019) Enhanced semantic visual secret sharing scheme for the secure image communication. 79(23-24), 17057–17079.
2. Abraham AS, Nair LR, Deepa MS (2017) A novel method for evaluation of visual security of images. In: 2017 International Conference on Networks & Advances in Computational Technologies (NetACT). IEEE
3. Al-Khalid RI, Al-Dallah RA, Al-Anani AM, Barham RM, Hajir SI (2017) A secure visual cryptography scheme using private key with invariant share sizes. J Softw Eng Appl 10(01):1–10.
4. Alex NS, Anbarasi LJ (2011) Enhanced image secret sharing via error diffusion in halftone visual cryptography. In: 2011 3rd International Conference on Electronics Computer Technology. IEEE
5. Amitharaja R, Shaik A, Thanikaise V (2016) Data security through data hiding in images: A review. J Artif Intell 10(1):1–21.
6. Arumugam S, Lakshmanan R, Nagar AK (2012) On  $(k, n)$ -visual cryptography scheme. 71(1), 153–162.
7. Askari N, Heys HM, Moloney CR (2013) An extended visual cryptography scheme without pixel expansion for halftone images. In: 2013 26th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE). IEEE
8. Askari N, Heys HM, Moloney CR (2014) Novel visual cryptography schemes without pixel expansion for halftone images. Can J Electr Comput Eng 37(3):168–177
9. Ateniese G, Blundo C, Santis AD, Stinson DR (1996) Visual cryptography for general access structures. Inf Comput 129(2):86–106.
10. Ateniese G, Blundo C, Santis AD, Stinson DR (2001) Extended capabilities for visual cryptography. Theor Comput Sci 250(1-2):143–161
11. Bhatnagar R, Kumar M (2018) Visual Cryptography: A literature survey. In: 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA). IEEE
12. Blesswin AJ, Visalakshi P (2013) An improved grayscale visual secret sharing scheme for visual information security. In: 2013 Fifth International Conference on Advanced Computing (ICoAC). IEEE
13. Bonis AD, Santis AD (2004) Randomness in secret sharing and visual cryptography schemes. Theor Comput Sci 314(3):351–374.
14. Borchert B (2007) Segment-based Visual Cryptography. Tech. Rep. WSI-2007-04, WSI Institute for computer science
15. Chanu OB, Neelima A (2018) A survey paper on secret image sharing schemes. International Journal of Multimedia Information Retrieval 8(4):195–215.
16. Chao HC, Fan TY (2017) Generating random grid-based visual secret sharing with multi-level encoding. Signal Process Image Commun 57:60–67
17. Chao H-C, Fan T-Y (2017) Random-grid based progressive visual secret sharing scheme with adaptive priority. Digital Signal Processing 68:69–80.
18. Chaturvedi A, Bhat IJ (2015) Analysis of schemes proposed for improving the segment based visual cryptography. International Journal of Computer Trends and Technology 30(1):26–30.
19. Chavan PV, Atique M (2012) Design of hierarchical visual cryptography. In: 2012 Nirma University International Conference on Engineering (NUICONe). IEEE
20. Chavan PV, Atique M (2015) Secured approach for authentication using threshold-based hierarchical visual cryptography. International Journal of Information Privacy, Security and Integrity 2(2):159.
21. Chavan PV, Atique M, Malik L (2014) Signature based authentication using contrast enhanced hierarchical visual cryptography. In: 2014 IEEE Students' Conference on Electrical, Electronics and Computer Science. IEEE
22. Chen J, Chen T-S, Hsu H-C, Lin Y-H (2009) Using multi-ringed shadow image of visual cryptography to hide more secret messages. The Imaging Science Journal 57(2):101–108.
23. Chen Q, Peng W-F, Zhang M, Chu Y-P (2010) An  $(n, n)$  threshold visual cryptography scheme for cheating prevention. In: 2010 3rd International Conference on Computer Science and Information Technology. IEEE
24. Chiu P-L, Lee K-H (2020) Threshold visual cryptography schemes with tagged shares. 8, 111330–111346.
25. Dahat AV, Chavan PV (2016) Secret sharing based visual cryptography scheme using cmy color space. Procedia Computer Science 78:550–555.
26. Das SS, Sharma KD, Chandra JK, Bera JN (2018) A hierarchical image cryptosystem based on visual cryptography and vector quantization. In: Advances in Intelligent Systems and Computing, pp 3–11. Springer Singapore
27. Deepa AK, Bento B (2014) Embedded extended visual cryptography scheme for color image using ABC algorithm. In: 2014 12th International Conference on Signal Processing (ICSP). IEEE
28. Dhiman K, Kasana SS (2018) Extended visual cryptography techniques for true color images. Computers & Electrical Engineering 70:647–658
29. Fang W-P (2008) Friendly progressive visual secret sharing. Pattern Recogn 41(4):1410–1414.
30. Fathimal M, Jansirani (2019) New fool proof examination system through color visual cryptography and signature authentication. In: The International Arab Journal of Information Technology, pp 322–336. Springer Berlin Heidelberg
31. Feng J-B, Wu H-C, Tsai C-S, Chang Y-F, Chu Y-P (2008) Visual secret sharing for multiple secrets. Pattern Recogn 41(12):3572–3581.
32. Fu Z, Yu B (2009) Research on rotation visual cryptography scheme. In: 2009 International Symposium on Information Engineering and Electronic Commerce. IEEE
33. Geetha P, Jayanthi VS, Jayanthi AN (2019) Multiple share creation based visual cryptographic scheme using diffusion method with a combination of chaotic maps for multimedia applications. 78(13), 18503–18530.
34. Guo Y, Jia X, Chu Q, Wang D (2020) A novel XOR-based threshold visual cryptography with adjustable pixel expansion. 10(4), 1321.

35. Haouzia A, Noumeir R (2007) Methods for image authentication: a survey. *Multimedia Tools and Applications* 39(1):1–46.
36. Hodeish ME, Humbe VT (2018) An optimized halftone visual cryptography scheme using error diffusion. *Multimedia Tools and Applications* 77 (19):24937–24953.
37. Hou YC, Lin CF, Chang CY (2001) Visual cryptography for color images without pixel expansion. *J Technol* 2(4):151.
38. Hou Y-C (2003) Visual cryptography for color images. *Pattern Recogn* 36(7):1619–1629.
39. Hou Y-C, Quan Z-Y (2011) Progressive visual cryptography with unexpanded shares. *IEEE Transactions on Circuits and Systems for Video Technology* 21(11):1760–1764.
40. Hou Y-C, Quan Z-Y, Tsai C-F, Tseng A-Y (2013) Block-based progressive visual secret sharing. 233, 290–304.
41. Hsu H-C, Chen J, Chen T-S, Lin Y-H (2007) Special type of circular visual cryptography for multiple secret hiding. *The Imaging Science Journal* 55 (3):175–179.
42. Hsu S-F, Chang Y-J, Wang R-Z, Lee Y-K, Huang S-Y (2012) Verifiable visual cryptography. In: 2012 Sixth International Conference on Genetic and Evolutionary Computing. IEEE
43. Ito KH, Tanaka H (1999) image size invariant visual cryptography. *IEICE transactions on fundamentals of electronics, communications and computer sciences* 22(10):3830–3841.