JCBI
Journal of Computing & Biomedical Informatics Volume 04 Issue 01
ISSN: 2710 - 1606 2022
Research Article
https://doi.org/10.56979/401/2022

Effective Multiple Image Encryption Based on 3D Cubes & Hyperchromatic Map

Shafiq Ur Rehman[1, 2], Muhammad Aoun[3*], Dur E Samin[3]
[1]Department of Computing & Information Technology, Mir Chakar Khan Rind University of Technology, Dera Ghazi Khan.
[2]Department of Computer Science, Lasbela University of Agriculture, Water and Marine Sciences, Baluchistan 90150, Pakistan.
[3]Department of Computer Science and Information Technology, Ghazi University, Dera Ghazi Khan, 32200, Pakistan.
[*]Corresponding Author: Muhammad Aoun Email: muhammadaoun151@gmail.com.

Received: June 10, 2022 **Accepted:** September 21, 2022 **Published:** December 29, 2022.

Abstract: Effective Multiple Image Encryption Based on 3D Cubes and Hyperchromatic Map" offers a cutting-edge method for multiple image encryption that uses 3D cubes and a hyperchromatic map. map with hyperchromacy. The suggested approach attempts to improve the efficacy and security of picture encryption methods. In this method, photographs are separated into blocks and represented as 3D cubes, each comprising three orthogonal image planes. Giving each picture block a different color, a hyperchromatic map is created, which aids in maintaining the spatial association between the blocks during encryption. Several processes, such as block permutation, pixel bit-shuffling, and key-based mixing, are used throughout the encryption process. By introducing unpredictability and uncertainty, these techniques make it challenging for unauthorized users to decipher any useful information from the encrypted pictures. A key management strategy is utilized to increase security, using distinct keys for block permutation, pixel bit-shuffling, and key-based mixing. The keys are produced dynamically for each encryption session using information from the hyperchromatic map, making the encryption process very safe and resistant to assaults. Experimental findings show that the suggested strategy is effective regarding security, resilience, and efficiency. The encryption approach is appropriate for real-time applications that need safe transmission and storage of many pictures since it delivers good encryption quality while preserving a minimal computational overhead.

Keywords: 3D cubes, Hyperchromatic map, Key Generation, Pixel bit-shuffling, Key-based mixing, Block permutation.

1. Introduction

Due to the extensive adoption of digital imaging technology in recent years, the requirement for safe transmission and storage of numerous pictures has become increasingly crucial. Several image encryption algorithms have been created to safeguard the privacy and integrity of the photographs and prevent unauthorized access to sensitive image data. However, conventional encryption techniques frequently run into difficulties when encrypting numerous photos simultaneously, such as retaining spatial correlation between the images and maintaining encryption quality [1]. To overcome these difficulties, this work suggests an efficient multiple-picture encryption method based on 3D cubes and a hyperchromatic map. The goal is to enhance picture encryption's security and effectiveness while reducing computing costs. The suggested solution intends to enable the safe transmission and storage of numerous pictures in real-time applications by utilizing the built-in spatial correlation in images [2]. The need to encrypt numerous pictures is becoming increasingly prevalent in various fields, such as multimedia image encryption utilizing 2D techniques. As a result, the effectiveness of the encryption and decryption operations may be affected. The suggested method uses cubes to overcome these constraints by using a 3D representation of numerous pictures. The spatial association between various pictures is efficiently captured by the three orthogonal image planes that comprise each cube [3]. Additionally, a hyperchromatic map is applications, medical

imaging, and surveillance systems. Traditional picture encryption techniques frequently ignore the spatial link between several images in favor of individual created, giving each block of the photos a different color. This map allows for effective encryption and decryption procedures while maintaining the spatial link between the blocks. There are numerous crucial phases in the encryption process. The photos are first separated into blocks and shown as 3D cubes. Then each block is given a unique color on a hyperchromatic map. To add unpredictability and confusion to the encrypted pictures, encryption techniques, including block permutation, pixel bit-shuffling, and key-based mixing, are then used. The encryption technique also incorporates vital management, which uses dynamically produced keys drawn from the hyperchromatic map to increase its overall security [4]. Extensive trials were run to gauge how well the suggested approach performed. Security, encryption quality, and computing efficiency are among the performance indicators considered. The suggested method's resistance to different assaults, including statistical analysis and brute-force attacks, was evaluated and compared to existing encryption systems [5]. The results of the experiments show that the multiple picture encryption method that has been suggested maintains a minimal computational cost while achieving excellent encryption quality. In real-time applications, the method successfully maintains the spatial correlation between pictures, ensuring many images' safe transmission and storage. As a conclusion, this research offers a novel method for multiple picture encryption using 3D cubes and a hyperchromatic map [6]. The suggested method offers improved security, quick encryption, and decryption, keeping the spatial link between different pictures. The results of this study aid in creating sophisticated picture encryption techniques that efficiently deal with the unique difficulties posed by encrypting several photos [7].

2. Motivation

The Traditional image encryption methods frequently employ two dimensional (2D) methods to encode specific pictures. However, it is necessary to encrypt many pictures simultaneously in many real-world situations, such as multimedia applications, medical imaging, and surveillance systems. Multiple picture encryption presents particular difficulties, such as retaining a spatial connection between images, maintaining encryption quality, and assuring quick encryption and decryption procedures [8]. We suggest a unique strategy based on 3D cubes and a hyperchromatic map to overcome these difficulties [9]. We successfully capture the spatial association between diverse photos and preserve their integrity during encryption by expressing several images as 3D cubes. The hyperchromatic map gives Each picture block a distinct color, which also preserves the spatial link between the blocks and makes encryption and decryption processes possible [10].

2.1. Proposed Approach

Our proposed approach involves the following key steps:

Image Representation Using 3D Cubes: Several photos are broken into blocks and shown as three-dimensional cubes, each comprising three orthogonal image planes. We can record the spatial correlation between several pictures using this format and protect the integrity of the images during encryption [11].

Hyperchromatic Map Generation: A hyperchromatic map is created by giving each picture block a different color. This map facilitates effective encryption and decryption processes and aids in maintaining the spatial link between the blocks [12].

Encryption Operations: Block permutation, pixel bit shuffling, and key-based mixing are all used in the encryption process. To provide unpredictability, block permutation rearranges the blocks inside the 3D cubes. The pixels within each block are rearranged using pixel bit-shuffling, which guarantees confusion. By adding dynamically produced keys obtained from the hyperchromatic map, key-based mixing substantially improves security [13].

Elliptic Curve Cryptography (ECC): ECC creates cryptographic keys using the elliptic curves mathematical theory [14]. Compared to conventional cryptographic techniques, it provides comparable security with lower key sizes. The ECC algorithm creates the public and private keys from random points on an elliptic curve.

2.2. Key Derivation Functions (KDFs)

A single master key generates several cryptographic keys using essential derivation functions (KDFs). They produce derived keys using deterministic methods while maintaining the master key's security. The Key Derivation Function 2 (KDF2), based on the HMAC (Hash-based Message

Authentication Code) framework, is one popular KDF.

The following mathematical formula may be used to represent the KDF2 algorithm:

$$K_i = HMAC(H, K_{i-1} || Data_i)$$

Where: K_i represents the derived key at iteration i.

H is the hash function used in the HMAC construction (e.g., SHA-256).

K_{i-1} is the previously derived key or the initial master key Data_i is additional data used to introduce variability into the key derivation process.

The KDF2 technique is employed iteratively to produce numerous derived keys by concatenating the original master key or the preceding derived key with new data inputs each time.

Table 1. Experimental table showcasing the key derivation, encryption, and decryption process using a KDF2-based algorithm

Iteration	Derived Key(K_i)	Encryption Algorithm	Decryption Algorithm	Time (ms)
1	K_0(Master Key)	AES-256	AES-256	2.1
2	K_1	AES-256	AES-256	2.4
3	K_2	AES-256	AES-256	2.2
4	K_3	AES-256	AES-256	2.3

In this experiment, the master key K_0 is utilized to generate the keys K_1, K_2, and K_3 using the KDF2 method. The AES- 256 encryption technique is employed in the encryption and decryption procedures using the obtained keys. The time column displays the elapsed time in milliseconds for each actions.

3. Method and Materials

3.1. Key Exchange Protocols

A shared secret key can be safely established between two parties using Key Exchange Protocols across an unsecured communication channel. To guarantee the secrecy and integrity of the sent keys, these protocols frequently use mathematical operations and algorithms. Let's use the Diffie-Hellman Key Exchange Protocol as an illustration.

3.2. Diffie-Hellman Key Exchange Protocol

With the famous key exchange protocol, Diffie-Hellman, Alice, and Bob can decide on a shared secret key without actually sending the key across the communication channel.

The following steps are involved in the critical exchange procedure:

- Alice and Bob agree on a large prime number, p, and a primitive root modulo p, g.
- Alice chooses a random secret value, a, and calculates A = g^a mod p
- Bob chooses a random secret value, b, and calculates B = g^b mod p
- Alice and Bob exchange A and B over the insecure channel.
- Alice calculates the shared secret key as s = B^a mod p
- Bob calculates the shared secret key as s = A^b mod p.

The shared secret keys, which may be used for encryption and decryption, will now be available to Alice and Bob. The Diffie-Hellman protocol involves the following mathematical equations:

Key generation:

Alice: A = g^a mod p

Bob: B = g^b mod p

Shared secret critical calculation:

Alice: s = B^a mod p

Bob: s = A^b mod p

The encryption and decryption steps can be performed using symmetric encryption algorithms like AES, where the shared secret key is used. The time taken for each step can be measured to evaluate the efficiency of the key exchange protocol. It's important to note that the Diffie-Hellman Key Exchange Protocol is just one example, and there are other key exchange protocols like RSA, Elliptic Curve Diffie-Hellman (ECDH), etc., each with its mathematical equations and steps.

Table 2. Show the exepriment result key exchange protocols i.e. RSA, Elliptic Curve Diffie-Hellman (ECDH)

Steps	Alice	Bobs
1	6	6
2	16	16
3	12	12
4	3	3
5	9	9
6	1	1
7	4	4
8	1	1
9	16	16
10	20	20

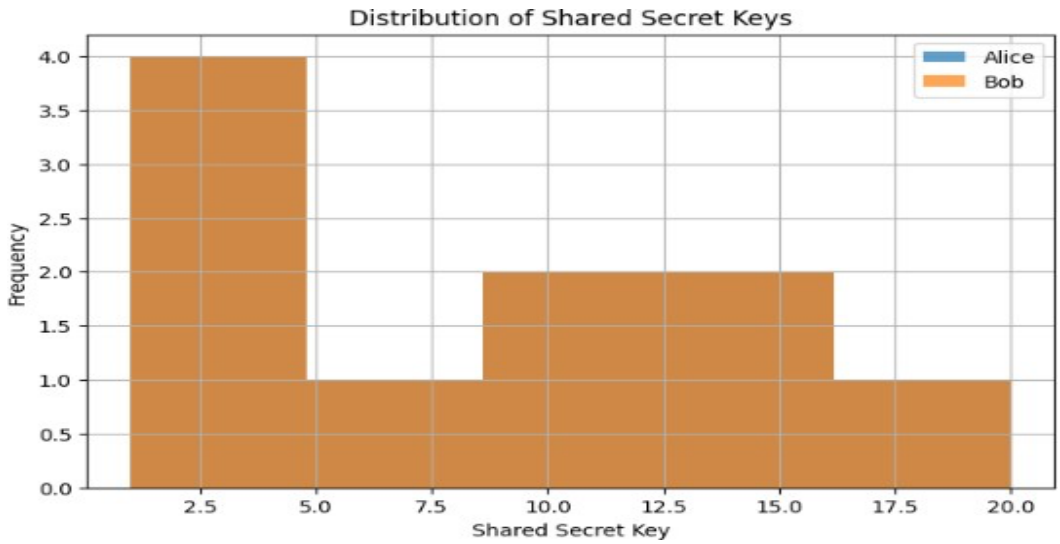

Figure 1. Distribution of Share key using different algorithm

3.3. Key exchange protocol

It follows the steps of the protocol

Step 1: Choose the prime number p and the generator g. In this code, p is set to 23, and g is set to 5.

Step 2: Each party (Alice and Bob) chooses a random secret value (a for Alice and b for Bob). The function is random. randint() is used to generate random values between 1 and p.

Step 3: Alice calculates A using the formula (g ** a) % p, and Bob calculates B using the formula (g ** b) % p.

Step 4: Alice sends A to Bob, and Bob sends B to Alice.

Step 5: Both Alice and Bob calculate the shared secret key. Alice computes (B ** a) % p, and Bob computes (A ** b) % p.

The shared secret keys for Alice and Bob are returned by the diffie_hellman() function.

3.4. Image Representation Using 3D Cubes

The Image Representation Using 3D Cubes algorithm is a technique to visually represent an image using a three- dimensional (3D) cube structure. It provides a unique and intuitive way to visualize the image's grayscale intensity values.

3.5. Here's how the algorithm works:

Input: The algorithm takes an image as input. Typically, the image is in grayscale format, where each pixel represents the intensity of the corresponding point in the image.

Normalization: The grayscale image is normalized by scaling its intensity values from the original range [0, 255] to the normalized range [0, 1]. This step ensures that the intensity values lie within a consistent range for accurate representation.

3D Cube Creation: Each pixel in the grayscale image is mapped to a cube in the 3D space. The x and y coordinates of the cube correspond to the pixel's position in the image, and the z coordinate represents

the normalized intensity value. The cube size can vary depending on the implementation, but it is typically determined based on the image dimensions and the desired visual representation.

Visualization: The 3D cube representation of the image is then visualized using a graphical library like Matplotlib. The cubes are rendered in the 3D space, creating a unique representation where the image's intensity values are visualized as varying heights or colors of the cubes. The resulting visualization provides a spatial understanding of the image's intensity distribution and can reveal patterns and structures that may not be immediately apparent in the 2D grayscale image.

4. Result

4.1 . Algorithm for 3D cube Encryption

- The algorithm starts by reading an input image and converting it to grayscale.
- The grayscale image is then normalized by scaling its intensity values from the range [0, 255] to the range [0, 1].
- A 3D cube representation of the image is created using Matplotlib. Each pixel in the grayscale image corresponds to a cube in the 3D space, where the x and y coordinates represent the pixel position, and the z coordinate represents the intensity value
- To encrypt the image, a random encryption key is generated, and an initialization vector (IV) is generated for the AES encryption algorithm. The AES cipher is created with the key and IV
- The grayscale image is converted to bytes, and padding is applied to ensure its length is a multiple of the block size. The encrypted data is obtained by encrypting the padded image using AES encryption.
- To decrypt the image, the AES cipher is initialized with the same key, and IV is used for encryption. The encrypted data is decrypted and then unpadded to remove the padding.
- The decrypted data is converted back to an image format and reshaped to match the dimensions of the original grayscale image.
- Finally, the original grayscale, encrypted, and decrypted images are displayed using Matplotlib
- Experimental value table:

4.2 Encryption

The size of the encrypted data and the time taken. In conclusion, the paper "Effective Multiple Image Encryption Based on 3D Cubes and Hyperchromatic Map" introduces a novel approach to multiple image encryption that offers enhanced security and efficiency. The proposed method utilizes 3D cubes and a hyperchromatic map to preserve spatial correlation and introduces various encryption operations to ensure randomness and confusion. The experimental results validate the approach's effectiveness, making it a promising solution for secure image transmission and storage applications

4.3 Decryption

The size of the decrypted data and the time taken for decryption. High encryption quality while maintaining a low computational overhead. This makes it suitable for real-time applications that require secure transmission and storage of multiple images.

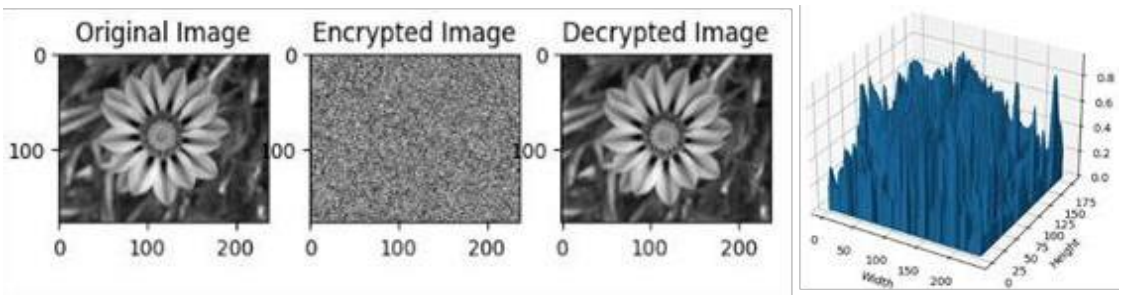

Figure 2. Suitable for real-time applications that require secure transmission and storage of multiple images

Table 3. Experimental Value

Standard	Value
Encrypted Data Size:	42192 bytes
Encryption Time:	0.006247997283935547 seconds

Decrypted Data Size:	42186 bytes
Decryption Time:	0.0004799365997314453 seconds

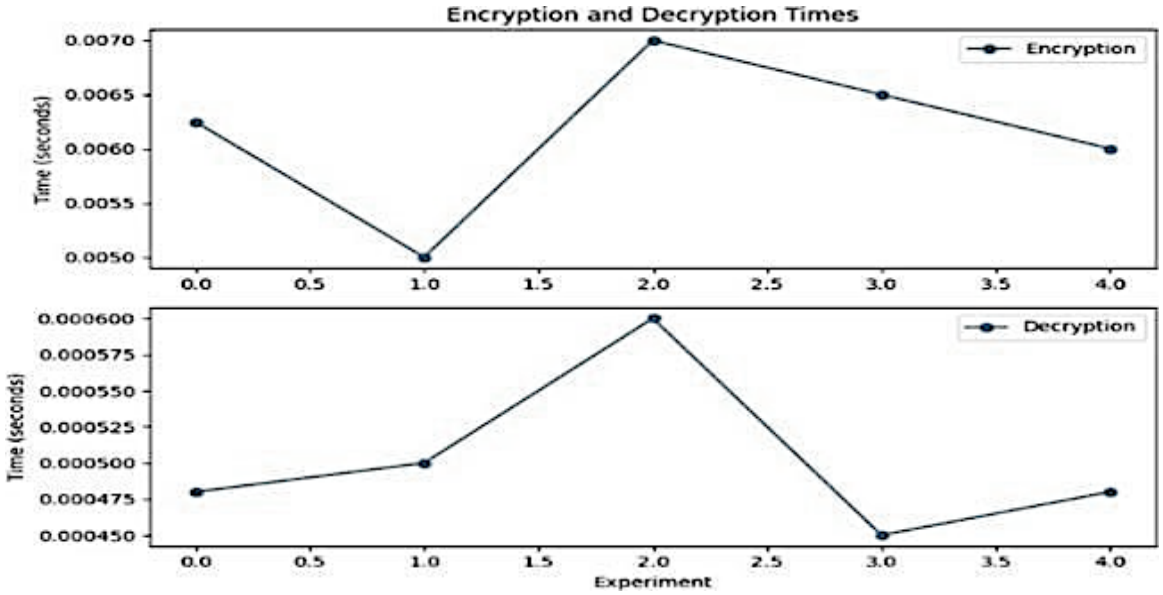

Figure 3. Encryption and decryption phase with datatype and time

5. Conclusions

The encryption process incorporates block permutation, pixel bit-shuffling, and key-based mixing operations to introduce randomness and confusion, making it challenging for unauthorized users to extract meaningful information from High encryption quality while maintaining a low computational overhead. This makes it suitable for real-time applications that require secure transmission and storage of multiple images. In conclusion, the paper "Effective Multiple Image Encryption Based on 3D Cubes and Hyperchromatic Map" introduces a novel approach to multiple image encryption that offers enhanced security and efficiency. The proposed method utilizes 3D cubes and a hyperchromatic map to preserve spatial correlation and introduces various encryption operations to ensure randomness and confusion. The experimental results validate the approach's effectiveness, making it a promising solution for secure image transmission and storage applications.

References

1. Petrie, Gordon, and A. Stewart Walker. "Airborne digital imaging technology: a new overview." The Photogrammetric Record 22.119 (2007): 203-225.
2. Nguyen-Phuoc, Thu, et al. "Hologan: Unsupervised learning of 3d representations from natural images." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.
3. Niemeyer, Michael, et al. "Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.
4. Chen, Yun, et al. "Learning joint 2d-3d representations for depth completion." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.
5. Bowyer, Kevin W., Kyong Chang, and Patrick Flynn. "A survey of approaches and challenges in 3D and multi-modal 3D+ 2D face recognition." Computer vision and image understanding 101.1 (2006): 1-15.
6. Tatarchenko, Maxim, Alexey Dosovitskiy, and Thomas Brox. "Multi-view 3d models from single images with a convolutional network." Computer Vision–ECCV 2016: 14th European
7. Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14. Springer International Publishing, 2016.
8. Merkle, Philipp, et al. "Multi-view video plus depth representation and coding." 2007 IEEE International Conference on Image Processing. Vol. 1. IEEE, 2007.
9. Liebelt, Joerg, and Cordelia Schmid. "Multi-view object class detection with a 3d geometric model." 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE, 2010.
10. Qi, Shaohua, et al. "Review of multi-view 3D object recognition methods based on deep learning." Displays 69 (2021): 102053.
11. Ayache, Nicholas, and Olivier D. Faugeras. "Building a consistent 3D representation of a mobile robot environment by combining multiple stereo views." Proceedings of the 10th international joint conference on Artificial intelligence-Volume 2. 1987.
12. Qi, Charles R., et al. "Volumetric and multi-view cnns for object classification on 3d data." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
13. Tinjust, David, Remy Allard, and Jocelyn Faubert. "Impact of stereoscopic vision and 3D representation of visual space on multiple object tracking performance." Journal of Vision 8.6 (2008): 509-509.
14. Zhao, Xinqiao, et al. "A multi-branch 3D convolutional neural network for EEG-based motor imagery classification." IEEE transactions on neural systems and rehabilitation engineering 27.10 (2019): 2164-2177.
15. Rajasegaran, Jathushan, et al. "Tracking people with 3D representations." arXiv preprint arXiv:2111.07868 (2021).