Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Securing High-Resolution Images from Unmanned Aerial Vehicles with DNA Encoding and Bit-Plane Extraction Method

ARSLAN SHAFIQUE[1], MUJEEB UR REHMAN [2], KASHIF HESHAM KHAN[3], SAJJAD SHAUKAT JAMAL[4], ABID MEHMOOD [5], AND SHEHZAD ASHRAF CHAUDHRY[5]

[1]Department of Electrical Engineering, Riphah International University (e-mail: Arslan.shafique@riphah.edu.pk)
[2]School of Science, Technology and Health, York St. John University, York, UK(e-mail: m.rehman@yorksj.ac.uk)
[3]School of Computing Technologies, STEM College, RMIT University (e-mail: KashifHesham.khan@rmit.edu.au)
[4]Department of Mathematics, College of Science, King Khalid University, Abha, Saudi Arabia(e-mail: shussain@kku.edu.sa)
[5]Department of Computer Science and Information Technology, College of Engineering, Abu Dhabi University, Abu Dhabi, UAE (e-mails: abid.mehmood@adu.ac.ae, ashraf.shehzad.ch@gmail.com)

Corresponding author: S.A. Chaudhry (e-mail: ashraf.shehzad.ch@gmail.com).

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University, for funding this work through a research group program under grant number R. G. P. 2/238/44

ABSTRACT Unmanned aerial vehicles (UAVs) are getting more popular for deployment in surveillance related operations owing to their flexibility and ability to reach hazardous areas. Moreover, the quality of digital cameras is getting better, and they can capture and store more visual information in high-resolution images. Unfortunately, due to the resource-constrained nature of UAVs, storing such large images can exhaust memory and related resources; whereas, the transmission of these images over the public link can pose several security threats. Securing these critical images during transmission from unauthorized access can be achieved through the use of efficient encryption techniques. This article proposes a novel encryption scheme incorporating both confusion and diffusion for encrypting both grey-scale and color images. In the proposed- encryption scheme, the image blocks are rearranged using a combination of random permutation, rotation, DNA encoding and zigzag pattern. Next, a bit-plane extraction method is used to obtain eight different bit-planes, including the most and least significant ones, from the scrambled image. These extracted bit-planes are then processed using confusion and diffusion techniques with a secret key, which is created using a hyper-chaotic map. The proposed method for encrypting the images taken by unmanned ariel vehicles is evaluated by examining its security level and time complexity using evaluation metrics such as correlation, entropy, energy, histogram analysis, keyspace and key sensitivity. The results and analysis demonstrated that the proposed encryption algorithm is able to effectively secure digital images. Additionally, the proposed work is also found to be superior to existing methods when compared using statistical security metrics.

INDEX TERMS Unmanned aerial vehicles, chaos theory, cyberattacks, DNA encoding

I. INTRODUCTION

UNMANNED aerial vehicles (UAVs), also known as Drones, are becoming more popular in a variety of industries for tasks such as security and surveillance. The drone sector is particularly thriving in sectors such as the military, and oil industries, where drones are used to inspect hazardous activities from the air and provide images for decision-making [1]. The importance of image encryption for drones is crucial due to their use in sensitive areas. However, it is essential that the encryption algorithms used are not only reliable and secure but also effective in terms of computational power [2]. Drones have several advantages over traditional ground-based surveillance methods. The use of drones in surveillance has become increasingly popular in recent years due to their numerous advantages over traditional ground-based security methods [3]. Drones have the ability to traverse large and hazardous areas discreetly and precisely, making them a cost-effective and compact alternative to manned aircraft. This also makes them harder to detect with radar systems. However, as drones continue to be equipped with advanced sensing and surveillance technology, the importance of securing them has become a crucial concern.

In recent years, information security has become a highly researched area due to the growth of digital information, such as images, videos, and audio. The transmission of this type of information over the Internet is challenging because of its bulk and high correlation compared to text, making traditional encryption algorithms like AES and DES unsuitable [4]–[8]. Although these algorithms are secure, they require a lot of computational time and are not practical for real-time applications [9]. The secure transmission of medical images is a critical issue, and researchers are exploring different approaches to encrypting images, including the Fourier Transform, Discrete Wavelet Transform, chaos and SCAN. Chaos-based image encryption algorithms have been found to be the most secure because of their complex structures, sensitivity to initial conditions, and ability to generate highly random sequences.

Chaos-based systems are commonly utilized to produce random key streams for image encryption through the selection of initial conditions and state variables, which are also referred to as key parameters or seeds [10]–[12]. These key streams are then used to encrypt images. A small change in the seed values can result in a completely different key stream, leading to significant differences in the encrypted image. In recent times, chaos-based image encryption has gained a lot of attention from researchers. Confusion-diffusion-based encryption algorithms are considered to be very secure, based on the concept introduced by Shannon. Confusion involves scrambling the pixels in several ways, such as row scrambling, direct pixel scrambling and column scrambling, while diffusion modifies the pixel values through transformations or logical methods [13]–[16]. In an image, pixels are the primary component and can consist of different numbers of bits. For example, in an 8-bit image, each pixel consists of 8 bits.

The level of security in an encryption scheme is improved when it encrypts information at the smallest unit. Bit-level encryption, which encrypts images on a bit-level basis, can result in a higher level of encryption compared to other methods [17], [18]. Bit-level encryption, confusion-diffusion-based and chaos-based encryption are all forms of spatial domain encryption. This type of encryption involves the direct manipulation of pixel values through mathematical operations such as substitution, and other logical operations. In contrast, frequency-domain encryption involves converting the pixel values into frequencies before manipulating them.

The way images are encrypted can be broken down into two main categories: spatial domain encryption and frequency domain encryption. In spatial domain encryption, the pixels of an image are directly manipulated using mathematical operations like permutation and substitution and logical operations like a cyclic shift. In contrast, frequency-domain encryption requires converting the pixel values of an image into frequencies, for example, through the use of a discrete wavelet transform. In this process, the original image is decomposed into four sub-bands, each consisting of different frequency components. The LL sub-band of an image holds a significant portion of the image's information [19]. In contrast, the other sub-bands (LH, HL, and HH) contain high-frequency information. By analyzing just the LL and LH sub-bands, it is possible to reconstruct the original image with minimal loss of information.

A. CONTRIBUTION OF THE WORK

Our aim in this work was to address the security concerns prevalent around the world by developing a solution to protect digital data from potential threats. The contributions of the paper include:

- In this study, we evaluated the existing encryption scheme and identified security vulnerabilities, which are detailed in sectionII. To ensure the protection of digital images, we designed an encryption technique that produces three different encrypted RGB images from one plaintext image. In order to access the plaintext information, the recipient must possess all three enciphered images.
- We propose an encryption method that combines different encryption methods such as bit-plane, chaos and DNA encoding. The design of the proposed encryption algorithm takes into account both the time required for the encryption process and the level of security it provides.
- In the proposed encryption method, we aim to minimize the computation time which can be obtained using the bit-plane extraction method. In the proposed work, only the most significant bi-planes (MSPs) are considered for encryption. The reason for choosing the MSPs is that the majority of the information of the original image is present in them. Therefore, it is more important to secure the MSPs compared to the least significant bit-planes (LSPs), as it might result in a time-consuming encryption process.
- We evaluate the effectiveness of the proposed encryption scheme through statistical analysis like PSNR, entropy, MSE, correlation and energy. Additionally, we tested the proposed algorithm against various attacks like noise attacks, cropping attacks and brute force attacks, to demonstrate its robustness against different cyber threats.

The rest of the paper is structured as follows: Section II presents a review of the existing encryption scheme and its limitations. Then, Section III offers a brief explanation of the secure drone monitoring architecture. Following that, Section IV covers the preliminaries that are employed in developing the proposed cryptosystem for high image resolutions. While section V and VI are devoted to the proposed cryptosystem and its experimental results and analysis, respectively. Finally, the paper concludes in Section VII and Section VIII presents the future work.

II. RELATED WORK

Chaos is often used in image encryption systems due to its unpredictable and sensitive nature. This means that even small changes in initial values can lead to vastly different outcomes [20], [21]. For example, Song et al. [22] utilized a chaotic system, specifically a one-dimensional skew-tent map, to encrypt color images. However, a chaotic system with low dimensions can be easily targeted by signal estimation algorithms, due to its straightforward chaotic path. Moreover, a low-dimensional chaotic map also has a small key space, making it vulnerable to cyberattacks. Hua et al. [23] presented an innovative framework based on exponential chaos in order to generate a secure and robust chaotic system. Similarly, Wang et al. [24] proposed a refined cross-coupled map lattice, which has high entropy and a larger chaotic range and applied it to encrypt the digital images. Such existing vulnerable encryption schemes demonstrate the ongoing efforts to improve chaotic systems for enhanced security. Furthermore, to improve security, various image encryption methods that integrate chaotic systems with other techniques have been proposed. Ghaffari et al. [25] utilized the measurement matrix produced by a Lorentz chaotic system to both encrypt and compress the original image. In [26], a four-wing hyperchaotic system is incorporated to generate DNA sequences dynamically, thereby diffusing the plaintext image. However, the existing encryption techniques are limited to encrypting a single image at a time and are not equipped to satisfy the demand for efficient transmission of information.

Encrypting multiple images at the same time is becoming a common trend in the field. There have been numerous new chaotic multi-image encryption methods developed in recent years. In [27], Man et al. introduced an effective image encryption scheme based on chaos and interpolation techniques. This allows for a single key to be used to encrypt multiple images. In [28] Shahna et al. proposed an image encryption algorithm that uses multiple colors and multiple levels of scrambling. The algorithm enhances the security of the encryption by linking the plain image's hash value with a cross-coupled PWLCM system. While Patro's technique enhances the performance of encryption, it does not address the issue of reducing storage space and transmission costs by compressing multi-plain images. This deficiency is also present in other encryption schemes such as [29], [30]. Recently, the development of compressive sensing (CS) based technology has allowed for simultaneous, compression, nonuniform sampling and encryption of digital images [31], [32]. One example is Luo et al.'s work [33], where the authors presented a compression, scrambling and diffusion-based strategy for image security. Their approach uses only a limited number of keys to generate the measurement matrix, which helps to minimize the storage space required for the keys. Additionally, Zhang et al. [34] introduced a unique dual-image encryption technique that combines double random encoding, diffusion-confusion and compressive sensing to enhance information transmission efficiency. Ye et al. [35] presented an optical multi-image encryption approach through the use of compressive sensing. In these schemes, the measurement matrix is made using a chaotic sequence. This lowers the cost of transmission and improves the performance of both compression and encryption. However, the encryption process in such schemes is not related to the characteristics of the natural image, and the same key streams are used for different plaintext images. This makes the encryption vulnerable to chosen plaintext attacks and can easily be broken.

Apart from the chaos and compress-sensing image encryption schemes, DNA computing is also applied to the design of encryption techniques that use dynamic rules to create secret keys. Digital images can be expressed as sequences of DNA bases [36]. Generally, DNA-based encryption works by converting the pixel values of an image into DNA bases and then performing encryption operations based on DNA operations. After the encryption process is completed, the DNA bases are transformed back into 8-bit pixel values [37]. Chaos has been integrated with DNA in several encryption algorithms [38], [39]. These algorithms were created to address the weaknesses of traditional image ciphers against cyberattacks and statistical attacks such as the entropy attack, chosen plaintext attack, and Bruce Force attack [40]–[42]. In [43], a combination of a chaotic logistic map and DNA is used to design an encryption algorithm. The chaotic logistic map controls the rules for the DNA encryption, and the secret key is derived from the plaintext image through a hash function. The results showed improved performance compared to other encryption methods; however, each image could only have a single secret key with a limited size.

In [44], a new encryption algorithm is introduced that combines DNA coding, chaotic maps, and arithmetic sequence scrambling. The algorithm uses the hash of the original image as the secret key, which is transformed into a chaotic sequence and then into DNA sequences for diffusion and confusion. Another encryption algorithm is proposed in [45] that uses a compound sine-piecewise linear chaotic map and varied DNA coding, with a focus on a straightforward design. While chaos is commonly used to secure plaintext images, there are concerns about their efficiency and security due to the computationally expensive floating-point operations and potential security weaknesses.

A. LIMITATIONS OF THE EXISTING ENCRYPTION SCHEMES

A few limitations of the existing encryption schemes mentioned in section IIare listed below:

1) **Low diffusion:** Chaotic systems can suffer from low diffusion, meaning that encrypted data may not be suf-

ficiently shuffled, which can result in residual patterns that could be exploited by an attacker.

2) **Lack of standardization:**There is no standardization of chaos-based encryption algorithms, which makes it difficult to compare and evaluate the security of different schemes.
3) **Vulnerability to known-plaintext attacks:** Some chaos-based encryption schemes are vulnerable to known-plaintext attacks, where an attacker can obtain information about the encryption key by using known plaintext and ciphertext pairs.
4) **Weaknesses in key generation:** A few existing encryption schemes heavily rely on the generation of secret keys, and any weaknesses or vulnerabilities in the key generation process can compromise the security of the encryption scheme.
5) **Dependence on the quality of randomness:** Encryption schemes depend on the quality of the randomness used in the system. If the randomness is not of high quality, the encryption scheme may be vulnerable to attacks.

III. SECURE AND SAFE DRONE MONITORING ARCHITECTURE

A secure drone monitoring framework refers to a set of protocols, systems, and tools that ensure the safe and secure operation of drones used for surveillance purposes. This framework would include elements such as encryption for sensitive data, secure communication channels for transmitting the data, and robust security measures to protect against hacking or unauthorized access [46]–[48]. Additionally, the framework might include guidelines for drone operation, such as specific flying patterns or altitude restrictions, to ensure the safety of both the drone and those in the area being surveyed. The goal of a secure surveillance drone framework is to provide a safe and reliable way to gather information while minimizing the risk of data breaches or other security incidents.

Drones have become a popular tool for surveillance due to their ability to capture live video of a specific area. They are especially useful in industries as they can access hazardous locations that would otherwise be difficult or even dangerous for human observers to reach. With a wide range of abilities to follow moving objects, drones provide a flexible and effective means of gathering information. However, drones also have some limitations, such as limited battery life and flight time, so they must be used strategically [49]–[51]. Due to the sensitive nature of drone footage, it must be encrypted before being transmitted via wireless.

The communication between a drone and its ground control station (GCS) is often not secure and requires encryption to protect sensitive data [52], [53]. To address this issue, a framework is needed that provides a secure means of transmitting the drone's images over potentially vulnerable communication channels. Drones are equipped with limited battery power and are capable of capturing high-quality images, making it essential that the encryption method used to be both fast and efficient. This will help to ensure that the drone can perform its surveillance operations optimally without being hampered by slow or resource-intensive encryption processes [54], [55]. To maximize the drone's performance, the encryption system must be designed with efficiency, taking into account factors such as processing time and the impact on the drone's battery life. Figure 1 shows the secure UAV monitoring framework, designed specifically for sensitive areas, consisting of the UAV, the GCS, and the communication link. However, an attacker may be able to intercept the insecure communication link and carry out a "man-in-the-middle" attack, which is why encryption is necessary. The framework proposed in this paper utilizes an image encryption algorithm based on bit-plane extraction and DNA encoding methods to securely transmit drone data.

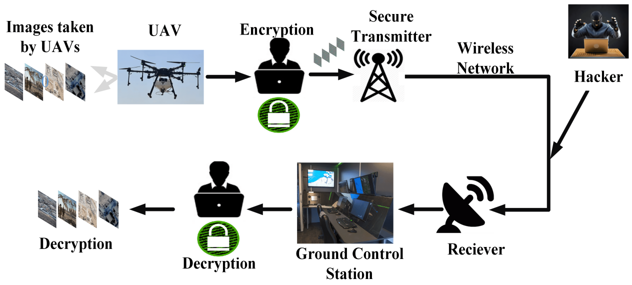


FIGURE 1: Secure UAV monitoring framework

IV. PRELIMINARIES

In this section, a preliminary overview of Deoxyribonucleic acid (DNA)-based encryption, hyper-chaotic maps and bit-plane extraction-based encryption is given, as they are crucial components of the proposed encryption scheme. All of these ideas are central to the encryption method, and a clear understanding of them is essential for comprehending the proposed algorithm and its purpose.

A. DEOXYRIBONUCLEIC ACID

Nucleic acids are tiny bio-polymers that are essential for the survival of all known forms of life. DNA and RNA, the two types of nucleic acids, are made up of nucleotides. These nucleotides contain four bases: cytosine (C), thymine (T), guanine (G) and adenine (A) in DNA, and the three bases (A, C, and G) are the same in RNA, but the fourth base, thymine, is replaced by uracil (U) [56]. The bases have complementary partners; for instance, A is complementary to T and C is complementary to G. A sequence of nucleotides can be represented by a symbolic string, where the biopolymers are expressed using symbols.

The utilization of DNA concepts in the field of computing, specifically in data encryption, has become widespread because of its simplicity in transforming binary numbers into nucleotides for processing [57]. Each of the four nucleotides can be symbolized using two bits of information, specifically the combinations 00, 01, 10, and 11, with eight different mappings of these bits to nucleotides available (as shown in Table 1). Four different types of operations can be done on such four nucleotides, and the XOR operation is frequently employed in DNA-based encryption schemes. The results of the XOR operation on all possible combinations of DNA nucleotides are presented in Table 2. The DNA-XOR operation is a straightforward and efficient logical operation that is widely utilized in encryption as the DNA-XOR operation makes it possible to retrieve the original plaintext after decryption has taken place [58], [59]. This operation also has the benefit of producing an equal number of 0s and 1s, which makes it a good choice for making the key schedule and encryption algorithm proposed in this paper.

00	A	C	A	C	G	T	G	T
01	G	C	A	T	A	T	G	C
10	T	C	G	T	A	G	A	C
11	A	T	C	G	G	T	A	C

TABLE 1: DNA encoding rule

XOR	T	A	C	G
A	T	A	C	G
C	G	A	G	T
T	C	G	A	C

TABLE 2: XOR-DNA encoding rule

B. 2D HYPER-CHAOTIC MAP

The 2D hyper-chaotic map (HCM) is a technique used to shuffle the positions of pixels in a plain image [60]. This method is based on a 2D hyper-chaos discrete nonlinear dynamic system and it can be represented as follows:

$$\begin{cases} A_{n+1} = b_1 + b_2 A_n + b_4 r_n \\ r_{n+1} = a_1 + a_3 A_n^2 \end{cases} \tag{1}$$

where b_1=0.3; b_2=0.4; b_4=0.6; a_1=1.6; a_3=3.8. The general representation of the system is given below:

$$\begin{cases} A_{n+1} = g(A_n, r_n) \\ r_{n+1} = f(A_n, r_n) \end{cases} \tag{2}$$

Where;

$$g(A_n, r_n) = b_1 + b_2 A_n + b_3 A_n^2 + b_4 r_n + b_5 r_n^2 + b_6 A_n r_n, b_i \in R, i = 1, 2, \cdots, 6$$

$$f(A_n, r_n) = a_1 + a_2 A_n + a_3 A_n^2 + a_4 r_n + a_5 r_n^2 + a_6 A_n r_n, a_i \in R, i = 1, 2, \cdots, 6$$

The Lyapunov exponent of the hyperchaotic system is converted into (A_n, r_n)

$$\begin{pmatrix} \partial_{A n+1} \\ \partial_{r n+1} \end{pmatrix} = \begin{pmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{pmatrix} \begin{pmatrix} \partial_{A n} \\ \partial_{r n} \end{pmatrix}$$

where;

$$f_{11} = \frac{\partial g}{\partial A} = b_2 + 2b_3 A_n + b_6 r_n;$$
$$f_{12} = \frac{\partial g}{\partial r} = b_4 + 2b_5 r_n + a_6 A_n;$$

$$\mathrm{f}_{21} = \frac{\partial f}{\partial A} = a_2 + 2a_3 A_n + b_6 r_n;$$
$$f_{22} = \frac{\partial f}{\partial r} = a_4 + 2a_5 r_n + a_6 A_n$$

Hyper-chaotic maps are used in the proposed work to generate secret keys in cryptography because they have high entropy and exhibit chaotic behavior, making them difficult to predict or reproduce without knowledge of the initial conditions. The basic idea is to use a hyper-chaotic map in the proposed work to generate a random sequence of numbers, which can then be used as a secret key for encryption and decryption. The sequence generated by the hyper-chaotic map is usually used to "seed" a pseudorandom number generator, which then generates the actual key used in the cryptographic system.

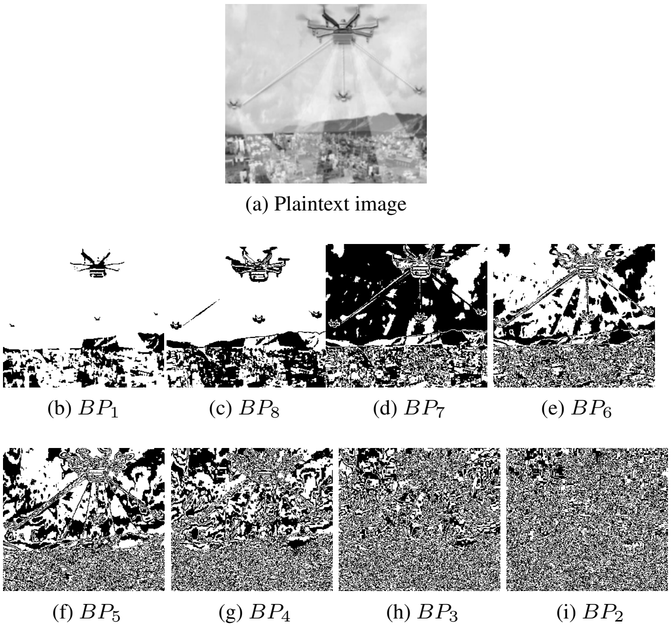
Equation 2 in the 2-D hyper-chaotic map is used to generate the random key sequences for permutation as well as diffusion purposes. The procedure for generating the random sequences using Equation 2 is given below:

- Specify the initial conditions, such as A_0, r_0, and f_0.
- Iterate Equation 2 65536 times to produce 65536 distinct random numbers in the sequence R, with each number ranging between 0 and 1 (i.e. $R \in (0, 1)$).
- The selection of initial values is performed on the bifurcation diagram of the hyperchaotic map. A detailed explanation of the bifurcation diagram of the hyperchaotic map and the criteria for selecting the initial key is given in [61].
- To enlarge the values of the sequence R, multiply each value with a large number say 999.
- Multiply each value in the sequence R by a large number, such as 999, to increase their values.
- To truncate the fractional values, apply the floor function or convert them to integers.
- if the plaintext image consists of 256 rows or columns, select the first 256 unique values from the random sequence R.
- Use the updated sequence R to scramble the pixel rows and columns of the plaintext image, corresponding to the confusion operation. To create diffusion, convert the entire 1-D sequence of 65536 different values into a 2-D matrix known as the $X - Mat$ of size 256 × 256, and then perform an XOR operation with the scrambled image.

  The generated sequence (R) and a 2-D matrix $(X - Mat)$ will be used as secrete key sequences.

C. BIT-PLANE EXTRACTION

Bit-plane (B-P) extraction is the process of separating an 8-bit digital image into 8 separate 1-bit images, each representing a single bit-plane of the original image [62]–[65]. Mathematically, the bit-planes $(B - P_j)$ can be extracted using Equation 3. This process can be accomplished using the following steps:



(a) Plaintext image

(b) BP_1 (c) BP_8 (d) BP_7 (e) BP_6

(f) BP_5 (g) BP_4 (h) BP_3 (i) BP_2

FIGURE 2: Eight bit-planes

$$\begin{aligned} B-P_1 &= (\frac{I_m}{2^0})mod(2), \quad B-P_2 = (\frac{I_m}{2^1})mod(2) \\ B-P_3 &= (\frac{I_m}{2^2})mod(2), \quad B-P_4 = (\frac{I_m}{2^3})mod(2) \\ B-P_5 &= (\frac{I_m}{2^4})mod(2), \quad B-P_6 = (\frac{I_m}{2^5})mod(2) \\ B-P_7 &= (\frac{I_m}{2^6})mod(2), \quad BP-8 = (\frac{I_m}{2^7})mod(2) \end{aligned} \tag{3}$$

Where I_m and 2^n represent the plaintext image and the position of the bit-plane respectively.

- Initialize a set of 8 1-bit images, each representing a single bit-plane of the original image.
- For each of the 8-bit-planes, isolate the corresponding bit from the original 8-bit image. This can be done using bit shifting and bit-wise operations.
- For each pixel in the original image, shift the corresponding 8-bit value right by the number of bits representing the bit-plane of interest.
- Mask the result with 1 (00000001 in binary) to obtain only the relevant bit. Store the resulting 1-bit value in the corresponding bit-plane image.
- Repeat steps 2–5 for each of the 8 bit-planes.
- The 8 resulting 1-bit images are the bit-planes of the original 8-bit image.

Bit-planes are categorized into two groups: most significant bits (MSBPs) and least significant bits (LSBPs). The information contained in each bit-plane varies. MSBPs possess the highest amount of information, while LSBPs have the lowest amount of plaintext information, as illustrated in Figure 2 [66], [67].

The quantity of plaintext information as a percentage can be computed using Equation 4, and the numerical values obtained are displayed in Table 3.

$$I_i = \frac{2^{j-1}}{\sum_{j=1}^{8} 2^{j-1}} \tag{4}$$

TABLE 3: Information percentage

BP_j **(j = 0, 1, · · · , 8)**	**Percentage values**
1	0.30
2	0.79
3	1.42
4	3.12
5	6.25
6	12.23
7	25.7
8	50.20

V. PROPOSED ENCRYPTION PROCESS

The proposed encryption method consists of four key steps. The first step involves scrambling the bitplanes that are extracted from the original plaintext image. The key generation process, which is based on a hyperchaotic map, is presented in the second stage. After the generation of the secret keys, a random image is generated, which is used for creating the discussion in the scrambled image using an XOR operation. The proposed encryption scheme utilizes symmetric keys, implying that the same secret key must be used by both the transmitter and the receiver. However, the receiver must have knowledge of the secret key information before receiving the encrypted image. The generalized process of the proposed encryption process is given in Figure 3.

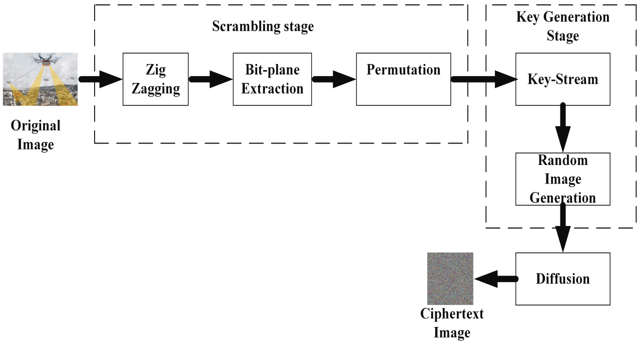
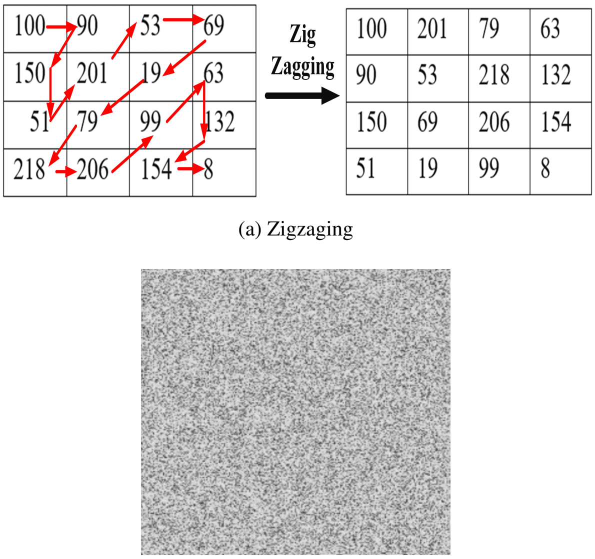


FIGURE 3: Different stages of the proposed encryption scheme

A. SCRAMBLING PROCESS

Before splitting the original image into its 8-bit planes, the original pixels value are shuffled as shown in Figure 4.

The original image pixel values are shuffled, as depicted in Figure 4, prior to being divided into 8-bit planes. As shown on the left side of Figure 4, a small portion of the plaintext image is selected as an example. Upon applying the Zigzag technique, a new image is generated, which can be seen on the right side of Figure 4. The pre-scrambled image is given in Figure 4(b).



100	90	53	69
150	201	19	63
51	79	99	132
218	206	154	8

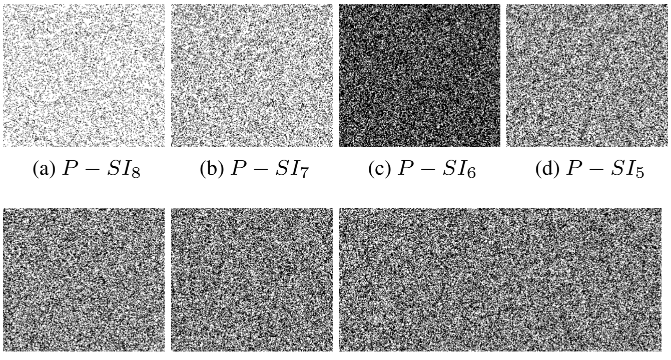
100	201	79	63
90	53	218	132
150	69	206	154
51	19	99	8

(a) Zigzaging

(b) Pre-scrambled image

FIGURE 4: Scrambling using zig-zagging process

Now extract the eight bit-planes from the pre-scrambled image $(P-SI)$ as shown in Figure 5 and shuffle the position of pixel rows and columns.


(a) $P-SI_8$ (b) $P-SI_7$ (c) $P-SI_6$ (d) $P-SI_5$

(e) $P-SI_4$ (f) $P-SI_3$ (g) $P-SI_2$ (h) $P-SI_1$

FIGURE 5: Bit-planes extracted from the pre-scrambled image

The mathematical of the permutation process which is applied to the $P-SI_i$ is given below:

Let take a portion of $P-SI$ (I):

$$I = \begin{bmatrix} 150 & 26 & 210 \\ 35 & 127 & 206 \\ 42 & 169 & 166 \end{bmatrix}$$

The binary version of $P-SI$ will be:

$$I_{bin} = \begin{bmatrix} 10010110 & 000110100 & 11010010 \\ 00100011 & 01111111 & 11001110 \\ 00101010 & 10101001 & 10100110 \end{bmatrix}$$

To extract the $P-SI_i$ from I_{bin}, consider each corresponding bit from each pixel. For instance, for the $P-SI_8$, choose 8^{th} bit from each pixel. Similarly, for the $P-SI_7$, choose 7^{th} bit from each pixel and so on. The extracted eight $P-SI_i$ form I_{bin} will be:

$$P-SI_8 = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, P-SI_7 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$, P-SI_6 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}, P-SI_5 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$P-SI_4 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}, P-SI_3 = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$, P-SI_2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}, P-SI_1 = 5 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

If we change the positions of values present in the $P-SI_i$, the permuted bit-planes $(PP-SI_8)$ will be:

$$PP-SI_8 = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}, PP-SI_7 = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

$$, PP-SI_6 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, PP-SI_5 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$PP-SI_4 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} PP-SI_3 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$, PP-SI_2 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}, PP-SI_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Following the permutation process, the binary values placed at position (1,1) in each PBP are merged to produce the eight binary values used for the permuted image's first-pixel value (1,1). Similarly, to acquire the (1,2) pixel in the permuted image, combine the binary values inserted at the location (1,2) in each P_BP. Repeat the same process to obtain all pixel values of the permuted image (P_{im}). The final permuted image corresponding to I_{BMI} will be:

$$P_{im} = \begin{bmatrix} 10010110 & 11111010 & 01010110 \\ 01100100 & 00100010 & 01111010 \\ 11001000 & 01001000 & 10101000 \end{bmatrix}$$

$$P_{im} = \begin{bmatrix} 150 & 250 & 86 \\ 100 & 34 & 122 \\ 200 & 72 & 168 \end{bmatrix}$$

The matrix P_{im} is generated by applying a permutation operation only on the MSBs. The resulting matrix is fully distinguishable from the original matrix I. Similarly, the whole process will be repeated for all the $P - SI_i$ in two iterations; (a) Row wise and (b) Column wise.

B. KEY GENERATION

The row and column-wise scrambling key generation process is given in algorithm 1.

Algorithm 1 Pseudo code for the generation of secrete keys and random image

```
Start

→ Input set the initial conditions for the hyperchaotic map. such as $a_0, b_0, r_0$ and $A_0$

→ Iterate the chaotic map one thousand times.
    for N = 1:1000
        $A_{n+1} = b_1 + b_2 A_n + b_4 r_n$
        $r_{n+1} = a_1 + a_3 A_n^2$

→ Enlarge the values generated in the sequence ($A$ and $r$)
        $A(n) = (A(n+1)) * 999$
        $r(n) = (r(n+1)) * 999$

→ Convert the fractional values into their decimal form.
        A(n) = floor(A(n))
        r(n) = floor(r(n))

→ Now take modulo of $A(n)$ and $r(n)$ with 256, to restrict the values between 0 to 255.
        $A_{mod}(n) = mod(A(n), 256);$
        $r_{mod}(n) = mod(r(n), 256)$
end
→ The Sequences $A_{mod}(n)$ and $r_{mod}(n)$ will be used for the row and column scrambling of $P - SI_i$, respectively.

→ To generate a random image, convert both the sequences ($A_{mod}(n)$ and $r_{mod}(n)$) into a two-dimensional array as follows:
        $A_{2D} = \text{reshape}(A_{mod}, row, col);$
        $r_{2D} = \text{reshape}(r_{mod}, row, col);$
End
```

C. DIFFUSION

In the diffusion process, the information contained in the image is altered by changing the values of each pixel. This results in the creation of a noisy image. To generate the final encrypted image, a substitution box and a bit-wise exclusive OR (XOR) operation is applied between the A_{2D} and the scrambled image data. A generalized pseudo-code of the proposed encryption algorithm is given in Algorithm 2. The figure presented in Figure 6 illustrates the detailed block diagram of the proposed work. Additionally, Figure 7 displays the test plaintext images and their corresponding ciphertext images. It is evident that the proposed encryption method can completely conceal plaintext information. This observation validates the efficacy of the proposed encryption scheme.

Algorithm 2 Pseudo code for the proposed encryption algorithm

```
Start
→Input Original images ($M_{im}$) of size Row = M × Columns = N
→$P_1$ = mod($M_{im}$, 2), $P_2$ = mod(floor($M_{im}$/4), 2), $P_3$ = mod(floor($M_{im}$/8), 2), $P_4$ = mod(floor($M_{im}$/16), 2), $P_5$ = mod(floor($M_{im}$/32), 2), $P_6$ = mod(floor($M_{im}$/64), 2), $P_7$ = mod(floor($M_{im}$/128), 2)  ▷ Bit-plane extraction: $P_1$, $P_1$, $P_1, \cdots P_7$ are bit-planes
→ Random sequence generation:
→ Iterate HCM for one thousand times
Setting the iteration to one thousand times is not compulsory and can be any large value capable of producing a minimum of 256 unique integer values.
→ Insert initial conditions and control parameters ($a_0$, $b_0$, $r_0$, $A_0$ in hyperchaotic map.
    for i = 1: 1000
        Iterate Equation 3        ▷ Result will store a variable called $Y_{i+1}$
        $Q(i)$ = $Y_{i+1}$ × 999
        $F(i)$ = floor($Q(i)$)
        $Md(i)$ = mod($F(i)$, 256)
    end
→ $X$ = unique ($Md(i)$, 256) ▷ Choose only unique values
→ Change initial conditions: ($a_0'$, $b_0'$, $r_0'$, $A_0'$)
→ Repeat all the above steps to generate another random sequence ($Y$) for column permutation
→ Row permutation:
        for row = 1:R
            rowperm($X$(R), :) = $M_{im}$(R, :);
        end
→ Column permutation:
        for col= 1:N
            colperm(:, $Y$(col1)) = rowperm(:, N);
        end
Create diffusion using XOR operation to generate a final encrypted image.
End
```

The proposed encryption scheme is ideal for securing drone images in comparison to other image encryption methods for the following factor and reasons.

- The structure of the proposed scheme is relatively straightforward when compared to other image encryption algorithms that use complex mathematical operations involving high-dimensional chaotic maps.
- The proposed encryption approach relies on the use of a Hyperchaotic map to create round keys and the final key,

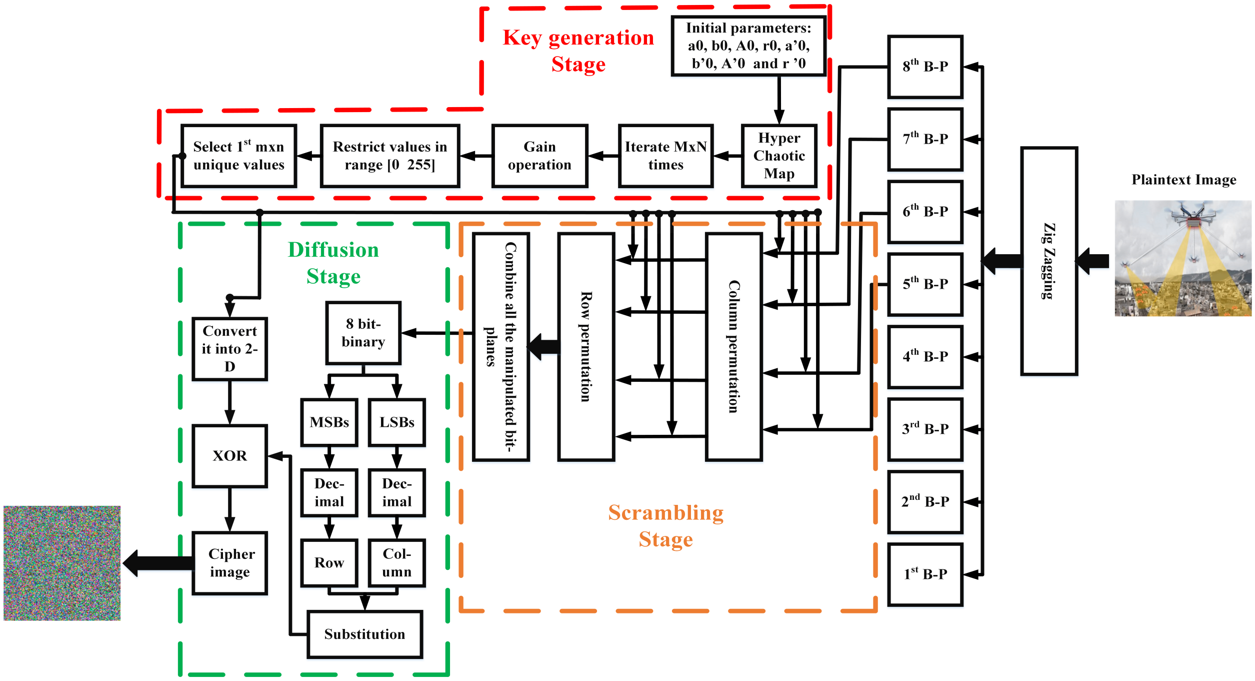


FIGURE 6: Detail block diagram of the proposed work

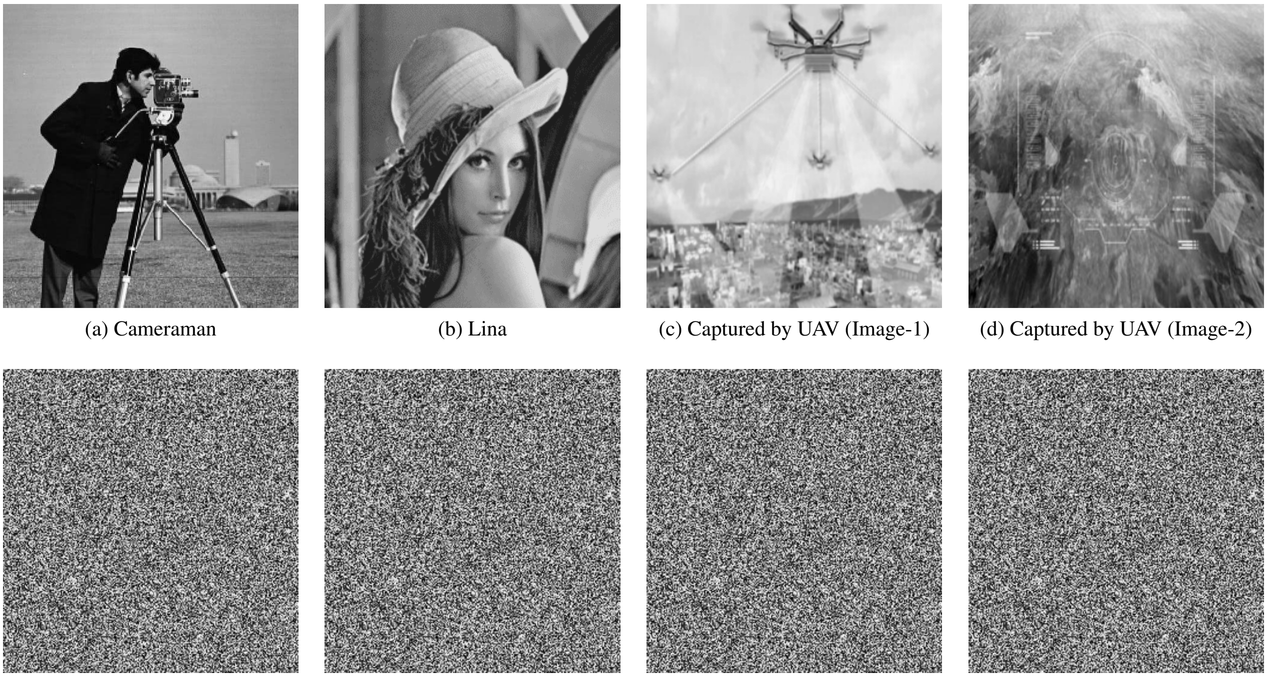

(a) Cameraman
(b) Lina
(c) Captured by UAV (Image-1)
(d) Captured by UAV (Image-2)
(e)
(f)
(g)
(h)

FIGURE 7: (a-d) Test image, (e-h) corresponding ciphertext images

which involves less computational effort and is simpler in nature.

- Lastly, the process of generating a random image through the Hyperchaotic map is crucial to producing the final key, which sets this encryption method apart from other chaos-based image algorithms that use complicated chaotic maps.
- The proposed method for encrypting images involves using the DNA bases of the pixels in the image along with round keys to create diffusion, which changes the

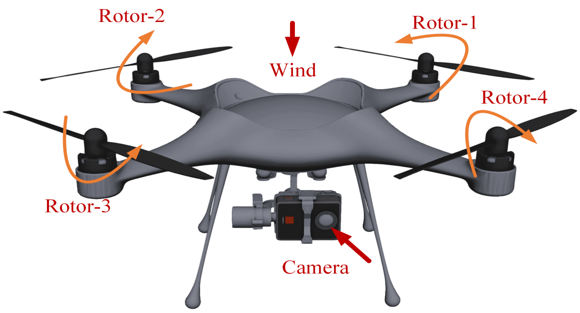


FIGURE 8: Multi rotor UAV

positions of the DNA bases in the image pixels.

- The proposed approach involves extracting bit-planes from the image and eliminating the least significant bit-planes from the encryption process, which helps to reduce the computational time required for encryption.
- This encryption method is particularly relevant for drone images, which can be very large and may contain sensitive information. Because drones have limited computational resources, the emphasis is on encrypting the images before sending them to the Ground Control Station (GCS) for decryption. The encryption algorithm needs to be designed with minimal latency, while the decryption algorithm can be performed on the GCS, which has more computational resources available.
- Compared to other nonlinear functions, DNA operations, chaotic operations, and bit-plane extraction are relatively simple and have lower computational complexity.

VI. EXPERIMENTAL RESULT AND ANALYSIS

The experiments in this section were carried out using MATLAB(R2022a) that makes processing matrices and signals simple and easy to implement. The computer used for the experiments had an Intel Core i5-1135G7 processor running at 2.40GHz, with 8GB of RAM and 11^{th}. To evaluate the efficiency of the proposed algorithm, it is also implemented a multi rotor UAV, as depicted in Figure 8. This drone has a maximum flight time of 34 minutes, can record 5K/90 fps videos, and capture 52-megapixel images. The images used in the experiments are of two types: standard images such as Lena and cameraman as shown in Figure 7, which are commonly used to evaluate the performance of modern encryption schemes, and images captured by a drone in an industrial zone for monitoring purposes. Such images have large sizes and high resolution.

A. HISTOGRAM ANALYSIS

A histogram is a graphical representation of the distribution of pixels in an image [68]. This information is useful when evaluating the strength of an encryption scheme. A strong encryption method should result in a histogram of the encrypted image that is uniform and flat, with no noticeable patterns compared to the original image. For example, Figure 9 shows a comparison of the histogram of a plaintext image and its corresponding encrypted images. The histogram of the ciphertext image shows a uniform distribution of pixels and appears to be completely different and lacks any recognizable patterns. This indicates that the proposed encryption scheme is able to resist attacks based on histogram analysis.

B. HISTOGRAM VARIANCE ANALYSIS

Variance is a statistical measure that is used to evaluate the uniformity of the histogram of encrypted images. Unlike histogram visualization, which provides a visual representation of the distribution of pixels, variance gives a numerical value that represents the spread of the data. It is often considered a more reliable metric for evaluating uniformity [69].

Variance can be calculated using a mathematical formula as shown in Equation 5 which gives a value that represents the spread of the data in the histogram. If the variance is low, it means that the data are tightly clustered around the mean, indicating a more uniform distribution. On the other hand, a high variance indicates that the data is spread out and less uniform.

$$Var(P) = \frac{1}{256}\sum_{K=1}^{256}[l_i - E(L)]^2 \qquad (5)$$

Where L is the image pixels stream, L = { l_1, l_2, l_3 ....,l_{256}, l_i is the image pixel value at K^{th} position and $E(L)$ = $\frac{1}{256}\sum_{K=1}^{256} l_i$.

Table 4 provides a comparison of the variance values of encrypted images produced by different encryption schemes. When analyzing the variance values, it becomes evident that the proposed encryption method outperforms the existing ones. The reason for this is that the variance values for the images produced by the new scheme are smaller, showing a more uniform distribution of pixels and providing a stronger form of encryption.

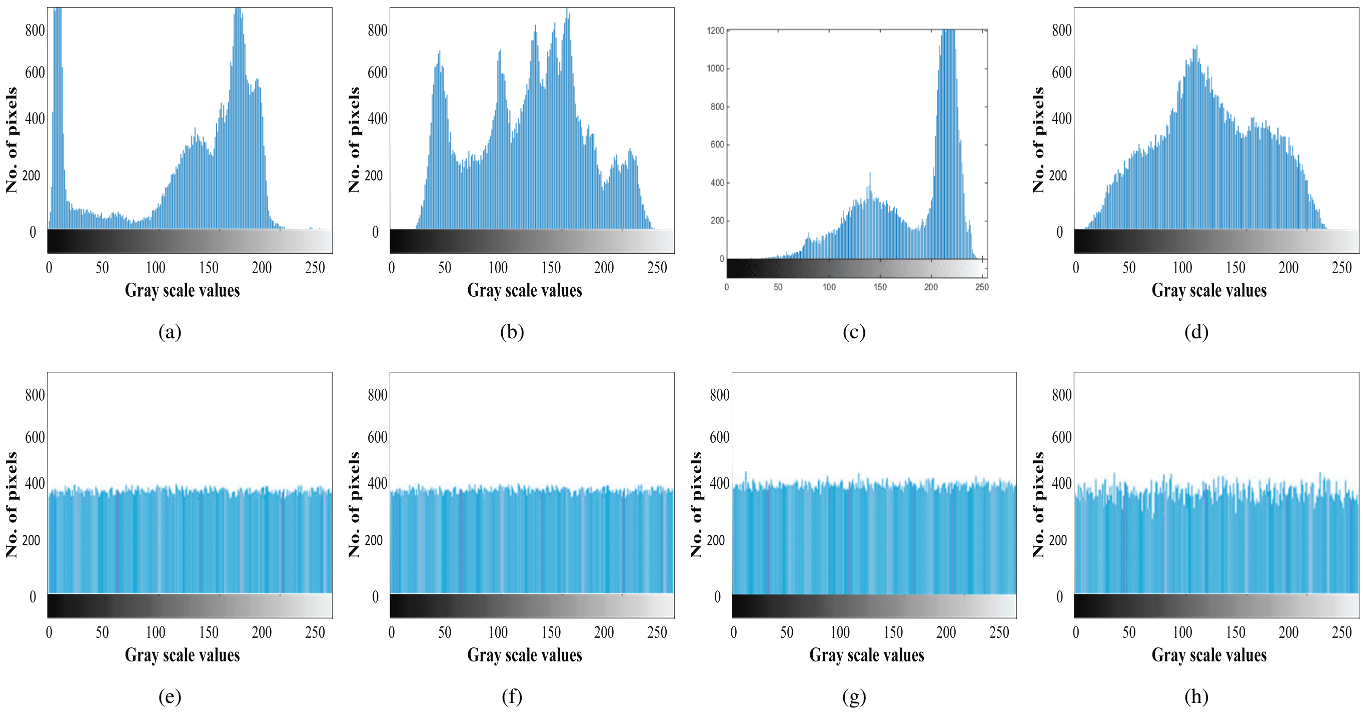
TABLE 4: **Histogram variance analysis**

Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Cameraman	271.17	272.17	268.33	277.53	255.63
Lina	261.97	263.78	271.72	274.94	262.60
Image$_1$	275.70	274.36	273.39	276.98	259.33
Image-1	271.37	274.36	276.87	274.31	269.96

C. MAXIMUM DEVIATION

The security of a cryptographic algorithm can be determined by the difference in pixel values between the original (plaintext) image and the encrypted (ciphertext) image. The greater the deviation in pixel values, the more secure the encryption technique is considered to be [74], [75].

Mathematically, this deviation can be expressed as the maximum difference between the image pixel values of the original and enciphered images as shown in Equation 6.



(a) (b) (c) (d)

(e) (f) (g) (h)

FIGURE 9: Histogram analysis: (a-d) Histogram of plaintext images, (e-h) histogram of corresponding ciphertext images

The larger this maximum deviation, the more robust the encryption scheme is in terms of security.

$$M_F = \frac{F_0 + F_{K-1}}{2} + \sum_{J=1}^{K-2} F_J \tag{6}$$

K and F_J represent the number of gray levels and range of histogram values at a specific index (J) respectively. If the value of M is high, it indicates that the encrypted image, known as the ciphertext, is notably distinct from the original image, referred to as the plaintext.

Table 5 shows the results of the maximum deviation for the proposed encryption method and other existing algorithms. By comparing the average values of the largest differences between the images, it can be determined that the proposed encryption algorithm and the methods mentioned in [66, 67] are better than other comparable encryption methods. This means that the amount of deviation in the proposed encryption method does not give away any important information about the encryption's strength. This indicates that the proposed encryption method is secure and does not reveal any important details of the original image.

TABLE 5: **Maximum deviation**

Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Camerman	24110	24897	25998	25631	24978
Lina	24993	24796	25689	25336	25789
Image$_1$	25196	25036	25687	25303	24231
Image$_2$base	24978	25639	25367	26031	25791

D. IRREGULAR DEVIATION

Measuring the encryption quality using just the maximum deviation (MD) is not sufficient. To determine the quality of the encryption, another metric known as Information Dissimilarity (ID) is employed. ID evaluates the encryption by measuring how similar the statistical distribution of the differences between the original and encrypted image is to a uniform distribution [76], [77]. This helps in assessing the evenness of the deviations between the two images.

The ID can be calculated using a mathematical formula given in Equation 7, which gives a value that represents the similarity of the deviation distribution to a uniform distribution. A lower value of ID indicates that the deviation distribution is closer to a uniform distribution, meaning that the encryption is stronger and more secure. On the other hand, a higher value of ID indicates that the deviation distribution is less uniform, meaning that the encryption is weaker and less secure.

$$I_d = \sum_{K=0}^{J-1} |T_K - C_G| \tag{7}$$

Where T_K represents the highest value in the histogram at position K, while C_G refers to the average of all the values in the histogram.

Table 6 displays the I_d values for the proposed and existing encryption techniques. The results demonstrate that the ID values for the proposed work are smaller compared to the other techniques, indicating that it has a higher encryption quality when compared to the others.

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3269294

TABLE 6: Irregular deviation

Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Cameraman	46978	47635	48569	47301	45031
Lina	45569	46996	47894	46687	45064
Image$_1$	46691	47630	47133	49963	45666
Image$_2$	46791	46698	47656	46687	45339

E. ENTROPY

In image encryption, entropy refers to a measure of the uncertainty or randomness of the image data. It provides information about the distribution of the image's pixels and can be used to assess the quality of the encryption process. The entropy of an encrypted image can be calculated as follows:

$$Entropy = -\sum C(P_j) log_2 C(P_j) \tag{8}$$

Where: P_j represents the probability of appearing in the j^{th} variable.

The maximum achievable entropy value in an encryption scheme relies on the image type being encrypted. For instance, in the case of an 8-bit image, the maximum attainable entropy value cannot exceed 8. In the proposed encryption technique, 8-bit images are considered for encryption. A value close to 8 indicates that the encryption scheme can provide a significant level of security to digital images. Table 7 presents the entropy values of different encrypted images and compares the outcomes with those of other existing encryption schemes. The findings demonstrate that the proposed encryption approach achieves entropy values that are much closer to the ideal value of 8 compared to other existing techniques, whose entropy values are lower than those of the proposed encryption method.

TABLE 7: Entropy analysis

Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Cameraman	7.9771	7.9853	7.9820	7.9835	7.9990
Lina	7.9865	7.9863	7.9965	7.9971	7.9991
Image$_1$	7.9896	7.9870	7.9923	7.9933	7.9990
Image$_2$	7.9799	7.9832	7.9834	7.9978	7.9989

F. CONTRAST

The contrast of an image refers to the difference in brightness between the lightest and darkest pixels in the image [78].

The primary goal of image encryption is to modify the contrast of the encrypted image such that it is dissimilar to the original image. This modification of contrast enhances the difficulty of extracting information from the encrypted image. The contrast of an image can be determined using Equation 9.

$$Contrast = \sum_{r,s=0} |r-s|^2 \gamma(r-s) \tag{9}$$

where r and s are 8-bit gray-level images, and $(r-s)$ is the gray-level occurrence matrix.

The contrast analysis involves calculating the difference in brightness between the pixels in the original image and the encrypted image, and measuring how different they are. If the difference is significant, it means the encryption process has changed the contrast of the image and has effectively concealed the original information. Table 8 shows the contrast values for the proposed encryption algorithm and other existing ones, where it can be seen that the contrast values for the proposed work are greater than the existing ones, which indicates that the proposed encryption method performs better than the other comparable encryption schemes.

TABLE 8: Contrast analysis

Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Cameramna	9.4986	9.5789	9.3689	9.8980	10.6989
Lina	9.7826	9.8910	9.7960	9.7741	10.2311
Image$_1$	9.8784	9.8812	9.7850	9.7713	10.6790
Image$_2$	9.8741	9.7780	9.8710	9.9820	10.3460

G. ENERGY

Energy analysis in image encryption is a measure of the randomness and uniformity of the pixel values in an encrypted image. The energy of an image is calculated as the sum of the squared pixel values of the image and is a measure of how much energy or "power" is present in the image. A higher energy value indicates that the pixel values are more uniformly distributed, while a lower energy value indicates that the pixel values are more clustered or unevenly distributed. The energy of an image can be calculated as:

$$Energy = \sum (i,j)^2 \tag{10}$$

This information can be used to assess the quality of an encryption scheme, as a good encryption scheme should produce encrypted images with small energy values, indicating that the pixel values are well randomized and the content of the original image is effectively concealed. Table 9 provides a comparison of the energy values for plaintext images, existing encryption methods, and the proposed encryption scheme. The outcomes demonstrate that the proposed encryption technique generates ciphertext images with energy values that are lower than those of both the plaintext images and other existing encryption methods.

TABLE 9: Energy analysis

Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Cameraman	0.0158	0.0159	0.0158	0.0164	0.0153
Lina	0.0164	0.0165	0.0161	0.0163	0.0151
Image$_1$	0.0160	0.0161	0.0162	0.0163	0.0154
Image$_2$	0.0160	0.0161	0.0159	0.0163	0.0153

H. LOSSLESS ANALYSIS

One critical aspect of an encryption algorithm is its capability to retrieve the original pixel values of the plaintext image with high accuracy, ensuring that the encryption process is lossless. Two commonly used metrics for assessing the lossless quality of an encryption algorithm are the peak signal-to-noise ratio (PSNR) and the mean square error (MSE), which can be mathematically represented as follows:

$$MSE = \frac{1}{KN}\sum_{v=0}^{K-1}\sum_{s=0}^{N-1}(P(v,s)-C(v,s))^2 \tag{11}$$

$$PSNR = 10 \times log_2 \frac{Q^2_{max}}{MSE} \tag{12}$$

Where, v and s are the row and column index respectively, $P(v,s)$ is plaintext image $C(v,s)$ is the ciphertext image and Q_{max} is the maximum plaintext image pixle value.

PSNR and MSE are two opposing measures used in image encryption. PSNR measures the similarity between the original image and the encrypted image, with higher values indicating a higher degree of similarity. This is not desirable in image encryption, where the goal is to have as little similarity as possible between the original and encrypted images. MSE, on the other hand, measures the differences between the two images. Therefore, the goal is to have the ciphertext image differ significantly from the plaintext image, to protect the image data. The proposed encryption algorithm is considered to be lossless if the PSNR is low and the MSE is high. The results reported in Table 10 and 11 for the proposed algorithm show a PSNR value of zero and an MSE value of infinity, respectively indicating strong encryption. Meanwhile, the existing encryption algorithms have PSNR and MSE values other than zero and infinity, indicating that they are not suitable for applications where the exact pixel values are needed to be retrieved.

TABLE 10: PSNR analysis

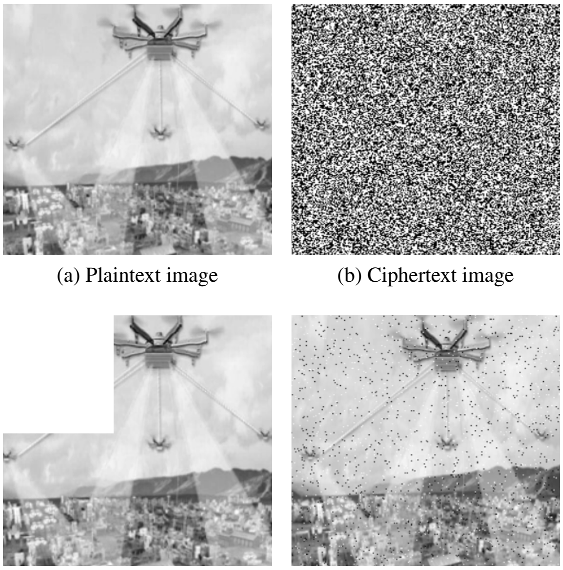
Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Cameraman	5.58	8.61	6.56	3.38	0
Lina	6.47	9.53	5.68	4.24	0
Image$_1$	3.47	8.34	4.26	3.56	0
Image$_1$	7.87	6.47	9.38	3.47	0

TABLE 11: MSE analysis

Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Camerman	201.81	204.81	186.31	195.41	∞
Linea	215.46	217.87	195.66	206.82	∞
Image$_1$	206.13	235.80	206.98	205.89	∞
Image$_2$	227.12	215.87	216.80	214.87	∞

I. CROPPEING ATTACK ANALYSIS

In order to test whether the proposed encryption scheme can withstand a cropping attack, a portion of the ciphertext image, with dimensions of 250×250, is cropped. The resulting cropped image is then input into the decryption algorithm to visualize if the original information can still be retrieved. From Figure 10(d), it can be observed that after the plaintext image is decrypted from the cropped ciphertext image, the original information can still be visualized, with some distortion. This demonstrates that the proposed encryption scheme is capable of withstanding a cropping attack and can successfully recover the original information even from a portion of the encrypted image.



(a) Plaintext image

(b) Ciphertext image

(c) Cropped ciphertext image

(d) Recovered image from cropped ciphertext image

FIGURE 10: Cropping attack analysis

J. NOISE ATTACK ANALYSIS

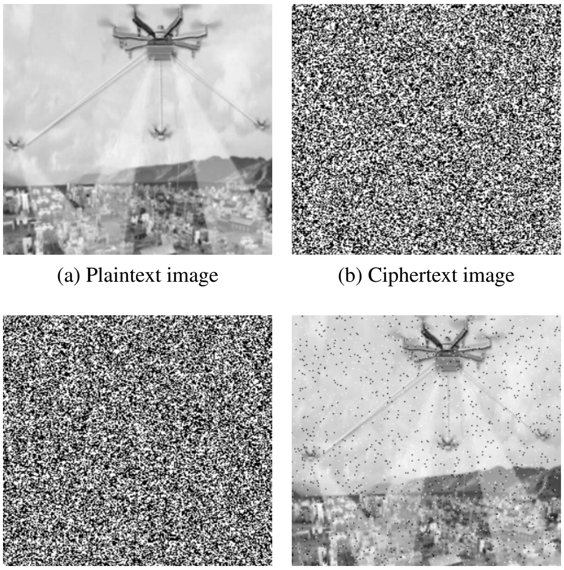
To test how well an encryption algorithm can withstand a noise attack, the algorithm is first used to encrypt an image. Then, to simulate a noise attack, random noise is added to the encrypted image using XOR operation as shown in Equation 13.

$$Noisy_{image} = Noise \oplus C(i,j) \tag{13}$$

Where $\oplus$ and $C(i,j)$ represent the XOR operation and the ciphertext image respectively. Following the addition of noise to the ciphertext image, the proposed decryption algorithm is utilized to perform decryption. Figure 11 displays the decrypted image, which indicates that the image's contents can be easily visualized. Although the decrypted image contains some noise, the differences between the decrypted image and the original image are negligible.

K. COMPUTATIONAL TIME ANALYSIS

Computational time analysis is an essential aspect of image encryption since it measures the time required to perform encryption and decryption operations on an image. It is particularly crucial for real-time drone applications, such as video streaming, where images need to be encrypted and decrypted quickly.



(a) Plaintext image (b) Ciphertext image

(c) Noisy ciphertext image (d) Recovered image from noisy ciphertext image

FIGURE 11: Noise attack analysis

The computational time for image encryption depends on several factors, including the encryption algorithm's complexity, the size of the image, and the hardware used for encryption. More complex encryption algorithms tend to take longer to encrypt an image, while larger images also require more time to encrypt or decrypt. Additionally, the type and processing power of the hardware used for encryption can significantly impact the computational time required.

The proposed work involved measuring the computational time of the encryption method using a built-in MATLAB command called "tic-toc." Various images are taken using a drone camera and encrypted using the proposed encryption technique. Table 12 displays the numeric values of time taken to encrypt each image- of size 512 × 512 in seconds. The results indicate that the proposed encryption scheme can encrypt the plaintext images within a few milliseconds. Moreover, a comparison with existing encryption methods is also provided, showing that the proposed method is more efficient in terms of processing time.

TABLE 12: Computational time analysis (sec)

Plaintext images (512 × 512)	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Cameraman	1.026	0.879	0.346	0.099	0.0002
Lina	0.986	0.846	0.647	0.0642	0.0006
Image_1	0.964	0.571	0.648	0.0698	0.0009
Image_2	1.015	0.945	0.206	0.087	0.0005

L. CORRELATION

In image encryption, correlation refers to the statistical relationship between two or more variables or attributes in an image. The correlation between two variables can be positive, negative, or zero. A positive correlation denotes that an increase in one variable leads to an increase in the other variable, while a negative correlation indicates that an increase in one variable leads to a decrease in the other variable. A zero correlation indicates that there is no relationship between the two variables.

The correlation between the original and encrypted images can be used to assess the quality of the encryption process. A low correlation between the two images indicates stronger encryption, as it means that the encrypted image is significantly different from the original image and that the encryption has effectively concealed the original image data. Mathematically, correlation can be represented as follows:

$$\text{CorrCoff} = \frac{Cov(e,r)}{\sigma_e \sigma_r}, \qquad \sigma_e = \sqrt{VAR_e}, \qquad \sigma_r = \sqrt{VAR_r}$$
$$\text{VAR (m)} = \frac{1}{C}\sum_{v=1}^{R}(\text{m}_s \text{ - E(m)})^2, \qquad \text{Cov(m, k)} = \frac{1}{C}$$
$$\sum_{v=1}^{R}(\text{m}_s \text{ - E(m) (g}_s \text{ - E(k)}$$

The symbols E and 'σ' stand for the expected value operator and the standard deviation, respectively

The correlation between the pixels in the original image is typically high, making it easy to see the content. However, a properly encrypted image will have less correlation between the pixels, making it more difficult to visualize the content. The goal is to have a low correlation value in the encrypted image so that the original image data is effectively concealed. Table 13 displays the results of the correlation analysis of the original images, existing encryption methods, and the proposed encryption method. It can be seen from the table that the proposed encryption method results in a significantly lower correlation value compared to both the original image and other existing methods.

TABLE 13: Correlation analysis

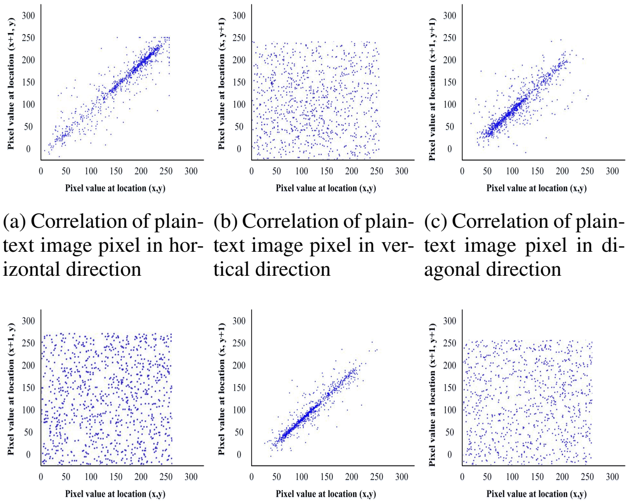
Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Cameraman	0.006	0.0025	0.0028	0.0011	0.0001
Lina	0.0029	0.0021	-0.0121	-0.0036	-0.0017
Image_1	0.0013	-0.0011	-0.0025	0.0011	-0.0001
Image_2	0.0017	-0.0025	-0.0013	-0.0013	-0.0028

In addition to statistical analysis, visual analysis can also be used to assess the correlation between image pixels of the plaintext and ciphertext images by plotting the scattered diagram as shown in Figure 12.

Figure 12(a-c) illustrates the scatter plot of the pixels in the original image in the horizontal, vertical, and diagonal directions. The dots on the plot, representing the pixels, are clustered closely together, indicating a high level of correlation between the image pixels. In contrast, Figure 12(d-f) shows the scatter plot of the pixels in the encrypted image in all three directions, and we can observe that the dots are widely separated from each other. This demonstrates that the proposed encryption technique effectively eliminates the correlation between the pixels in the image.

M. DIFFERENTIAL ATTACK ANALYSIS

The encryption process should exhibit a high level of sensitivity to even the slightest variations in the input. Even



(a) Correlation of plaintext image pixel in horizontal direction

(b) Correlation of plaintext image pixel in vertical direction

(c) Correlation of plaintext image pixel in diagonal direction

(d) Correlation of ciphertext image pixel in horizontal direction

(e) Correlation of ciphertext image pixel in vertical direction

(f) Correlation of ciphertext image pixel in diagonal direction

FIGURE 12: Scattered plots of plaintext and ciphertext image pixels

a small modification in one pixel's value should produce a significant shift in the texture of the output image. As the difficulty level increases in establishing the relationship between the input and output images, the effectiveness of differential cryptanalysis decreases. To show the robustness against differential attack analysis, two well-known metrics including the number of pixels' change rate (NPCR) and unified average changing intensity (UACI), are used. For effective encryption, the statistical values of these metrics should be as high as possible.

1) NPCR and UACI

This analysis considers a pair of encrypted images that originate from source images differing by only one pixel. The NPCR and UACI can be calculated using Equation 14 and 15, respectively.

$$NPCR = \frac{\sum_{s,t} D(s,t)}{R \times C} \times 100\% \tag{14}$$

$$UACI = \frac{1}{R \times C} \sum_{s,t} \frac{|E_1(s,t) - E_2(s,t)|}{255} \times 100\% \tag{15}$$

Where R and C denote the rows and columns of pixels in the image, and E_1(s,t) and E_2(s,t) refer to two ciphertext images whose corresponding plaintext images differ by only a single pixel. The matrix $D(s,t)$ can be determine as follows:

$$D(s,t) = \begin{cases} \text{if } \; E_1(s,t) = E_2(s,t), & 0 \\ \text{if } \; E_1(s,t) \neq E_2(s,t), & 1 \end{cases}$$

Tables 14 and 15 display the UACI and NPCR values for both the proposed and existing encryption schemes. Upon analyzing these values, it becomes evident that the proposed encryption scheme outperforms the existing one and exhibits greater resistance against differential attacks.

TABLE 14: UACI

Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Cameraman	33.3550	33.4972	33.5225	33.3131	33.6654
Lina	33.4950	33.5172	33.5025	33.5731	33.6454
Image$_1$	33.5049	33.5172	33.5325	33.4231	33.6537
Image$_2$	33.5138	33.5272	33.5525	33.5731	33.6618

TABLE 15: NPCR

Plaintext images	Ref [70]	Ref [71]	Ref [72]	Ref [73]	Proposed
Cameraman	99.6132	99.6234	99.6345	99.6123	99.6634
Lenna	99.6531	99.6341	99.6220	99.6451	99.6653
Image$_1$	99.6345	99.6310	99.6421	99.6451	99.6612
Image$_2$	99.6437	99.6275	99.6189	99.6034	99.6679

N. CHOSEN PLAINTEXT ATTACK

To obtain the symmetric key K_c (it can be correct or incorrect) in this attack, a plaintext image is generated with specific characteristics as follows:

1) Create an image consisting of all ones except for the first (4×4) pixels, which are represented by: $\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$
2) The values of the initial (2×2) pixels, identified as A11, are known despite being randomly chosen.
3) Solving the following equation results in obtaining the remaining (4 x 4) first pixels:

$$\begin{cases} P_{12} = I - P_{11} \\ P_{21} = I - P_{11} \\ P_{11} + P_{11} = 0 \end{cases} \tag{16}$$

4) Let E represents the (4×4) pixels of the encrypted image.
5) By taking the modulo of (2^8 or 2^{16}), the shared secret key can be obtained as $K_c = (E \times P)$mod(2^8 or 2^{16})

By adopting the above procedure, a simulation of a chosen plaintext attack is performed.

1) Chosen plaintext attack simulation

The hyperchaotic map parameters are A_0 = 03.4678 and r_0 = 0.1345 this generates the key sequences for diffusion $(D - K_c)$ and random permutation $(RP - K_c)$ as given below:

$$(D - K_c)_{CPA} = \begin{bmatrix} 120 & 25 & 135 & 234 \\ 112 & 197 & 65 & 103 \\ 26 & 197 & 234 & 214 \\ 22 & 16 & 115 & 167 \end{bmatrix}$$

$$(RP - K_c)_{CPA} = \begin{bmatrix} 15 & 6 & 4 & 12 & 10 & 1 & 3 & 5 & 8 & 2 \\ 7 & 11 & 9 & 13 & 16 & 14 & & & & \end{bmatrix}$$

Let $P = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$

Then the chosen plaintext will be generated as follows:

$$P = \begin{bmatrix} 124 & 0 & 157 & 201 \\ 65 & 18 & 213 & 11 \\ 167 & 96 & 10 & 216 \\ 34 & 15 & 117 & 201 \end{bmatrix}$$

The encrypted matrix E corresponding to P will be:

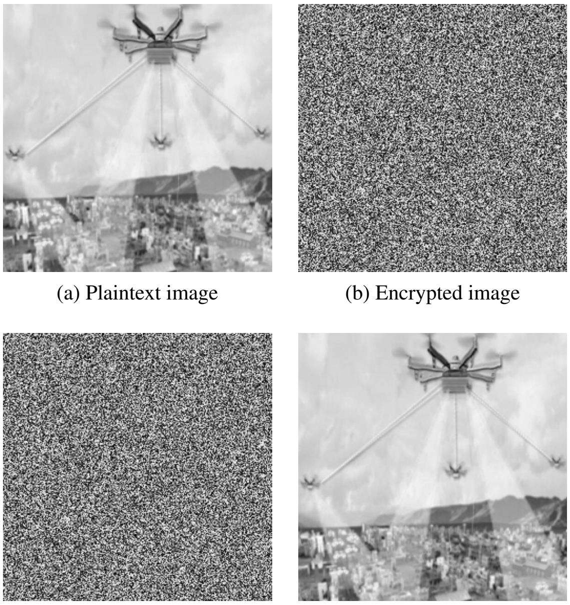
$$E = \begin{bmatrix} 25 & 19 & 102 & 211 \\ 36 & 271 & 0 & 254 \\ 112 & 137 & 10 & 69 \\ 94 & 27 & 156 & 117 \end{bmatrix}$$

Using the aforementioned results produced, the recovered diffusion key (R-D-K) and random permutation key (R-RP-K) secret will be as follows:

$$(R-D-K_c)_{CPA} = \begin{bmatrix} 95 & 59 & 112 & 154 \\ 230 & 19 & 97 & 73 \\ 210 & 157 & 10 & 86 \\ 87 & 51 & 167 & 201 \end{bmatrix}$$

$$(R-RP-K_c)_{CPA} = \begin{bmatrix} 3, 8, 15, 1, 12, 9, 7, 2, 5, 10, 16, 11, 4, 6, 13, 14 \end{bmatrix}$$

The $R-D-K_c$ are $R-RP-K_c$ are not similar to $D-K$ and $RP-K$, respectively. Therefore, as depicted in Figure 13, it is nearly impossible to accurately recover the original image.



(a) Plaintext image

(b) Encrypted image

(c) Recovered image using $(R-D-K)_{CPA}$ and $(R-RP-K)_{CPA}$

(d) Recovered image using $(D-K)_{CPA}$ and $(RP-K)_{CPA}$

FIGURE 13: Chosen plaintext attack simulation

O. KNOWN PLAINTEXT ATTACK

In certain situations, it may not be feasible to choose the plaintext image, but it can still be obtained. In such cases, the following steps can be followed to obtain the secret key K_c:

1) A window with dimensions of 4×4 is travelled over the plaintext image.
2) The invertibility of the modulo 2^8 or 2^{16} is checked for each window.
3) Let the invertible matrix of size 4×4, referred to as I_m be discovered, the secret key can be deduced using the following process:

$$K_c = E \times inv(I_m) mod(2^8 or 2^{16}) \quad (17)$$

The simulation of a Known plaintext attack is performed by adopting the procedure given in section VI-O1.

1) Known plaintext attack simulation

An invertible block of size (4×4) is discovered at position $(15, 19)$ in the original image illustrated in Figure 14(a), which is denoted as:

$$I_m = \begin{bmatrix} 112 & 146 & 116 & 178 \\ 164 & 146 & 120 & 179 \\ 110 & 113 & 145 & 102 \\ 134 & 167 & 102 & 167 \end{bmatrix}$$

The inverse of I_m with modulo 2^8 will be:

$$Inv(I_m) = \begin{bmatrix} 114 & 121 & 130 & 59 \\ 95 & 145 & 197 & 163 \\ 25 & 113 & 178 & 91 \\ 24 & 157 & 163 & 18 \end{bmatrix}$$

After performing the encryption, the encrypted matrix is given as:

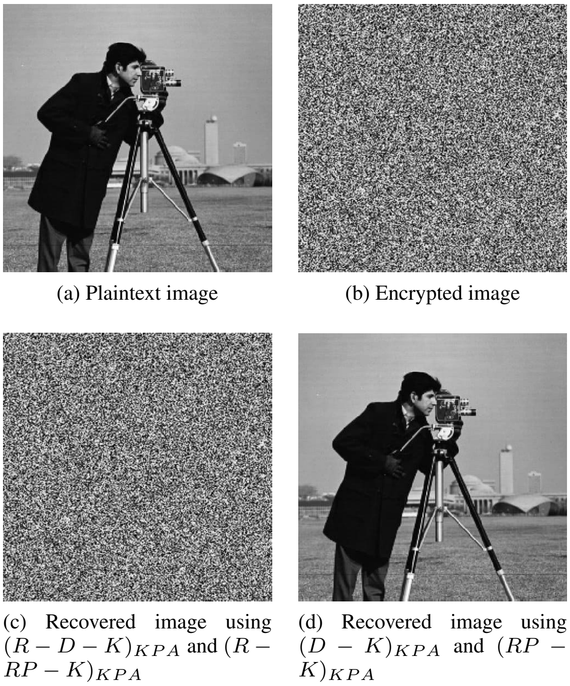
$$E = \begin{bmatrix} 154 & 210 & 15 & 98 \\ 114 & 167 & 201 & 251 \\ 154 & 169 & 137 & 85 \end{bmatrix}$$

Using Equation 17, the recovered diffusion and confusion keys are as follows:

$$(R-D-K_c)_{KPA} = \begin{bmatrix} 152 & 25 & 94 & 115 \\ 132 & 157 & 15 & 96 \\ 156 & 179 & 156 & 88 \\ 35 & 168 & 205 & 176 \end{bmatrix}$$

$$(R-RP-K_c)_{KPA} = \begin{bmatrix} 2, 15, 10, 7, 9, 6, 8, 13, 16, 11, 12, 1, 3, 14, 4, 5 \end{bmatrix}$$

By using $(R-D-K_c)_{KPA}$ and $(R-RP-K_c)_{KPA}$, it can be seen in Figure 14 that no information in the decrypted image which is recovered using the recovered keys can be visualized.



(a) Plaintext image

(b) Encrypted image

(c) Recovered image using $(R-D-K)_{KPA}$ and $(R-RP-K)_{KPA}$

(d) Recovered image using $(D-K)_{KPA}$ and $(RP-K)_{KPA}$

FIGURE 14: Known plaintext attack simulation

VII. CONCLUSION AND DISCUSSION

This study presents a novel approach to secure image encryption for drone-based surveillance systems. The researchers have developed a key scheduling algorithm that combines elements of DNA and a hyperchaotic map. The secret key is utilized to generate multiple round keys, which can be adjusted to different sizes. The scrambling process is carried out using the bit-plane extraction method, and the diffusion operation is based on the integer representation of these bases and the XOR operation. The design of the encryption algorithm aims to provide both confusion and diffusion properties in a single round, ensuring a high level of security. The algorithm and its key schedule underwent extensive statistical analysis. The results of these experiments indicate that the encryption keys are randomly generated, and the encryption process is efficient, even when applied to large images. The proposed algorithm has demonstrated that it can withstand any type of malicious attack. It has met the necessary spatial demands for drone usage, however, there is still room for improvement in terms of the speed of encryption for large images. To overcome this issue, future work can focus on making the proposed encryption scheme even simpler to enhance its overall efficiency.

While the proposed encryption scheme has not been tested in an actual UAV scenario, we have implemented it in MATLAB and conducted a series of experiments to analyze its real-time performance. For instance, we used the "Tic-Toc" command, which operates in real-time, to calculate the total time required for encrypting high-resolution images, and we evaluated the scheme's strength using various security parameters. These assessments demonstrate that the proposed encryption technique is robust enough to withstand cyberattacks.

In a practical situation involving unmanned aerial vehicles (UAVs), our analysis was conducted using MATLAB in real-time. The performance of the encryption technique we proposed is influenced by the processing speed of the UAV's processor. Consequently, if a fast processor is employed in the UAV, our encryption method is likely to operate more quickly and encrypt data in a more time-efficient manner.

Apart from that, Unlike existing schemes, where pixel-level permutation is used, which involves permuting image pixels based on DNA or zigzag pattern techniques, the proposed work achieves permutation at the bit-level. This involves decomposing the image into its bit planes and then permuting the bits within the bit-planes using random sequences generated using DNA and chaotic maps. The use of bit-level permutation instead of pixel-level permutation results in better security and time efficiency, as shown in Tables 4-13 in the manuscript.

VIII. FUTURE WORK

The proposed work can be integrated with the existing approaches given in [79], [80]. One way to achieve this is by utilizing the memristor-based neural network circuit [79] to intelligently encrypt high-resolution images. By incorporating this approach, the strength of the proposed encryption scheme and other existing schemes can be detected within seconds. Additionally, the use of DNA chaotic synchronization for secure communication [80] can aid in identifying the region of interest in high-resolution images that require encryption. By adopting these techniques, the encryption computational time can be reduced even further while also enhancing the security strength.

Conflict of Interest:

- The authors have no conflicts of interest
- This manuscript is neither submitted nor is under review at any other journal for publication.

REFERENCES

[1] Q. Jiang, H. Bai, and X. He, “Design of robust sensing matrix for uav images encryption and compression,” Applied Sciences, vol. 13, no. 3, p. 1575, 2023.

[2] F. Syed, S. K. Gupta, S. Hamood Alsamhi, M. Rashid, and X. Liu, “A survey on recent optimal techniques for securing unmanned aerial vehicles applications,” Transactions on Emerging Telecommunications Technologies, vol. 32, no. 7, p. e4133, 2021.

[3] J. A. Reis-Filho and T. Giarrizzo, “Drone surveys are more efficient and cost effective than ground-and boat-based surveys for the inspection of fishing fleet at harbors,” Coasts, vol. 2, no. 4, pp. 355–368, 2022.

[4] A. Shafique and F. Ahmed, “Image encryption using dynamic s-box substitution in the wavelet domain,” Wireless Personal Communications, vol. 115, pp. 2243–2268, 2020.

[5] A. Shafique, “A new algorithm for the construction of substitution box by using chaotic map,” The European Physical Journal Plus, vol. 135, no. 2, p. 194, 2020.

[6] A. Shafique and J. Shahid, “Novel image encryption cryptosystem based on binary bit planes extraction and multiple chaotic maps,” The European Physical Journal Plus, vol. 133, no. 8, p. 331, 2018.

[7] B. Langenberg, H. Pham, and R. Steinwandt, "Reducing the cost of implementing the advanced encryption standard as a quantum circuit," IEEE Transactions on Quantum Engineering, vol. 1, pp. 1–12, 2020.

[8] D. E. Standard et al., "Data encryption standard," Federal Information Processing Standards Publication, vol. 112, 1999.

[9] A. Shafique, J. Ahmed, W. Boulila, H. Ghandorh, J. Ahmad, and M. U. Rehman, "Detecting the security level of various cryptosystems using machine learning models," IEEE Access, vol. 9, pp. 9383–9393, 2020.

[10] M. B. Farah, R. Guesmi, A. Kachouri, and M. Samet, "A novel chaos based optical image encryption using fractional fourier transform and dna sequence operation," Optics Laser Technology, vol. 121, p. 105777, 2020.

[11] R. Ge, G. Yang, J. Wu, Y. Chen, G. Coatrieux, and L. Luo, "A novel chaosbased symmetric image encryption using bit-pair level process," IEEE Access, vol. 7, pp. 99470–99480, 2019.

[12] M. Gafsi, N. Abbassi, M. A. Hajjaji, J. Malek, and A. Mtibaa, "Improved chaos-based cryptosystem for medical image encryption and decryption," Scientific Programming, vol. 2020, pp. 1–22, 2020.

[13] X. Gao, "Image encryption algorithm based on 2d hyperchaotic map," Optics Laser Technology, vol. 142, p. 107252, 2021.

[14] M. U. Rehman, A. Shafique, S. Khalid, and I. Hussain, "Dynamic substitution and confusion-diffusion-based noise-resistive image encryption using multiple chaotic maps," IEEE Access, vol. 9, pp. 52277–52291, 2021.

[15] H. Tora, E. Gokcay, M. Turan, and M. Buker, "A generalized arnold's cat map transformation for image scrambling," Multimedia Tools and Applications, vol. 81, no. 22, pp. 31349–31362, 2022.

[16] A. Shafique, A. Mehmood, and M. Elhadef, "Survey of security protocols and vulnerabilities in unmanned aerial vehicles," IEEE Access, vol. 9, pp. 46927–46948, 2021.

[17] X. Wang, S. Lin, and Y. Li, "Bit-level image encryption algorithm based on bp neural network and gray code," Multimedia Tools and Applications, vol. 80, pp. 11655–11670, 2021.

[18] A. Hasheminejad and M. Rostami, "A novel bit level multiphase algorithm for image encryption based on pwlcm chaotic map," Optik, vol. 184, pp. 205–213, 2019.

[19] A. Shafique, J. Ahmed, M. U. Rehman, and M. M. Hazzazi, "Noiseresistant image encryption scheme for medical images in the chaos and wavelet domain," IEEE Access, vol. 9, pp. 59108–59130, 2021.

[20] Rupa, Ch, et al. "Securing multimedia using a deep learning based chaotic logistic map." IEEE Journal of Biomedical and Health Informatics (2022).

[21] Nagasree, Yarajarla, et al. "Preserving Privacy of Classified Authentic Satellite Lane Imagery Using Proxy Re-Encryption and UAV Technologies." Drones 7.1 (2023): 53.

[22] Y. Song, J. Song, and J. Qu, "A secure image encryption algorithm based on multiple one-dimensional chaotic systems," in 2016 2nd IEEE international conference on computer and communications (ICCC), pp. 584–588, IEEE, 2016.

[23] A. H. Bukhari, M. A. Z. Raja, N. Rafiq, M. Shoaib, A. K. Kiani, and C.-M. Shu, "Design of intelligent computing networks for nonlinear chaotic fractional rossler system," Chaos, Solitons & Fractals, vol. 157, p. 111985, 2022.

[24] M. Wang, X. Wang, T. Zhao, C. Zhang, Z. Xia, and N. Yao, "Spatiotemporal chaos in improved cross coupled map lattice and its application in a bit-level image encryption scheme," Information Sciences, vol. 544, pp. 1– 24, 2021.

[25] A. Ghaffari, "Image compression-encryption method based on twodimensional sparse recovery and chaotic system," Scientific Reports, vol. 11, no. 1, p. 369, 2021.

[26] E. Dong, M. Yuan, S. Du, and Z. Chen, "A new class of hamiltonian conservative chaotic systems with multistability and design of pseudorandom number generator," Applied Mathematical Modelling, vol. 73, pp. 40–71, 2019.

[27] Z. Man, J. Li, X. Di, Y. Sheng, and Z. Liu, "Double image encryption algorithm based on neural network and chaos," Chaos, Solitons Fractals, vol. 152, p. 111318, 2021.

[28] K. Shahna and A. Mohamed, "Novel hyper chaotic color image encryption based on pixel and bit level scrambling with diffusion," Signal Processing: Image Communication, vol. 99, p. 116495, 2021.

[29] A. Anees, A. M. Siddiqui, J. Ahmed, and I. Hussain, "A technique for digital steganography using chaotic maps," Nonlinear Dynamics, vol. 75, pp. 807–816, 2014.

[30] I. Hussain, A. Anees, A. H. AlKhaldi, A. Algarni, and M. Aslam, "Construction of chaotic quantum magnets and matrix lorenz systems sboxes and their applications," Chinese Journal of Physics, vol. 56, no. 4, pp. 1609–1621, 2018.

[31] X. Chai, J. Bi, Z. Gan, X. Liu, Y. Zhang, and Y. Chen, "Color image compression and encryption scheme based on compressive sensing and double random encryption strategy," Signal Processing, vol. 176, p. 107684, 2020.

[32] A. Shafique, "A noise-tolerant cryptosystem based on the decomposition of bit-planes and the analysis of chaotic gauss iterated map," Neural Computing and Applications, vol. 34, no. 19, pp. 16805–16828, 2022.

[33] Y. Luo, Y. Liang, S. Zhang, J. Liu, and F. Wang, "An image encryption scheme based on block compressed sensing and chen's system," Nonlinear Dynamics, pp. 1–21, 2022.

[34] R. Zhang and D. Xiao, "A secure image permutation–substitution framework based on chaos and compressive sensing," International Journal of Distributed Sensor Networks, vol. 16, no. 3, p. 1550147720912949, 2020.

[35] G. Ye, C. Pan, Y. Dong, K. Jiao, and X. Huang, "A novel multi-image visually meaningful encryption algorithm based on compressive sensing and schur decomposition," Transactions on Emerging Telecommunications Technologies, vol. 32, no. 2, p. e4071, 2021.

[36] S. E. El-Khamy and A. G. Mohamed, "An efficient dna-inspired image encryption algorithm based on hyper-chaotic maps and wavelet fusion," Multimedia Tools and Applications, vol. 80, pp. 23319–23335, 2021.

[37] A. Anees and I. Hussain, "A novel method to identify initial values of chaotic maps in cybersecurity," Symmetry, vol. 11, no. 2, p. 140, 2019.

[38] S. Zhou, "A real-time one-time pad dna-chaos image encryption algorithm based on multiple keys," Optics Laser Technology, vol. 143, p. 107359, 2021.

[39] M. Alawida, J. S. Teh, D. P. Oyinloye, M. Ahmad, R. S. Alkhawaldeh, et al., "A new hash function based on chaotic maps and deterministic finite state automata," IEEE Access, vol. 8, pp. 113163–113174, 2020.

[40] A. Anees, "An image encryption scheme based on lorenz system for low profile applications," 3D Research, vol. 6, pp. 1–10, 2015.

[41] A. Anees, A. M. Siddiqui, and F. Ahmed, "Chaotic substitution for highly autocorrelated data in encryption algorithm," Communications in Nonlinear Science and Numerical Simulation, vol. 19, no. 9, pp. 3106–3118, 2014.

[42] A. Shafique, M. M. Hazzazi, A. R. Alharbi, and I. Hussain, "Integration of spatial and frequency domain encryption for digital images," IEEE Access, vol. 9, pp. 149943–149954, 2021.

[43] I. Aouissaoui, T. Bakir, and A. Sakly, "Robustly correlated key-medical image for dna-chaos based encryption," IET image processing, vol. 15, no. 12, pp. 2770–2786, 2021.

[44] X. Yan, X. Wang, and Y. Xian, "Chaotic image encryption algorithm based on arithmetic sequence scrambling model and dna encoding operation," Multimedia Tools and Applications, vol. 80, pp. 10949–10983, 2021.

[45] S. Zhang and L. Liu, "A novel image encryption algorithm based on spwlcm and dna coding," Mathematics and Computers in Simulation, vol. 190, pp. 723–744, 2021.

[46] M. U. Rehman, A. Shafique, Y. Y. Ghadi, W. Boulila, S. U. Jan, T. R. Gadekallu, M. Driss, and J. Ahmad, "A novel chaos-based privacypreserving deep learning model for cancer diagnosis," IEEE Transactions on Network Science and Engineering, vol. 9, no. 6, pp. 4322–4337, 2022.

[47] A. Shafique, A. Mehmood, and M. Elhadef, "Detecting signal spoofing attack in uavs using machine learning models," IEEE access, vol. 9, pp. 93803–93815, 2021.

[48] A. Shafique, A. Mehmood, M. Elhadef, and K. H. Khan, "A lightweight noise-tolerant encryption scheme for secure communication: An unmanned aerial vehicle application," Plos one, vol. 17, no. 9, p. e0273661, 2022.

[49] J.-P. Yaacoub, H. Noura, O. Salman, and A. Chehab, "Security analysis of drones systems: Attacks, limitations, and recommendations," Internet of Things, vol. 11, p. 100218, 2020.

[50] M.-A. Lahmeri, M. A. Kishk, and M.-S. Alouini, "Artificial intelligence for uav-enabled wireless networks: A survey," IEEE Open Journal of the Communications Society, vol. 2, pp. 1015–1040, 2021.

[51] M. A. Khan, B. A. Alzahrani, A. Barnawi, A. Al-Barakati, A. Irshad, and S. A. Chaudhry, "A resource friendly authentication scheme for space–air–ground–sea integrated maritime communication network," Ocean Engineering, vol. 250, p. 110894, 2022.

[52] S. A. Chaudhry, J. Nebhen, A. Irshad, A. K. Bashir, R. Kharel, K. Yu, and Y. B. Zikria, "A physical capture resistant authentication scheme for the internet of drones," IEEE Communications Standards Magazine, vol. 5, no. 4, pp. 62–67, 2021.

[53] M. W. Akram, A. K. Bashir, S. Shamshad, M. A. Saleem, A. A. AlZubi, S. A. Chaudhry, B. A. Alzahrani, and Y. B. Zikria, "A secure and lightweight drones-access protocol for smart city surveillance," IEEE Transactions

on Intelligent Transportation Systems, vol. 23, no. 10, pp. 19634–19643, 2022.

[54] A. Singh, S. C. Satapathy, A. Roy, and A. Gutub, "Ai-based mobile edge computing for iot: Applications, challenges, and future scope," Arabian Journal for Science and Engineering, pp. 1–31, 2022.

[55] S. Hussain, K. Mahmood, M. K. Khan, C.-M. Chen, B. A. Alzahrani, and S. A. Chaudhry, "Designing secure and lightweight user access to drone for smart city surveillance," Computer Standards Interfaces, vol. 80, p. 103566, 2022.

[56] D. Ravichandran, A. Banu S, B. Murthy, V. Balasubramanian, S. Fathima, and R. Amirtharajan, "An efficient medical image encryption using hybrid dna computing and chaos in transform domain," Medical & Biological Engineering Computing, vol. 59, pp. 589–605, 2021.

[57] K. Xuejing and G. Zihui, "A new color image encryption scheme based on dna encoding and spatiotemporal chaotic system," Signal Processing: Image Communication, vol. 80, p. 115670, 2020.

[58] T. A. Al-Maadeed, I. Hussain, A. Anees, and M. T. Mustafa, "A image encryption algorithm based on chaotic lorenz system and novel primitive polynomial s-boxes," Multimedia Tools and Applications, vol. 80, pp. 24801–24822, 2021.

[59] I. Hussain, A. Anees, M. Aslam, R. Ahmed, and N. Siddiqui, "A noise resistant symmetric key cryptosystem based on s 8 s-boxes and chaotic maps," The European Physical Journal Plus, vol. 133, pp. 1–23, 2018.

[60] A. G. Mohamed, N. O. Korany, and S. E. El-Khamy, "New dna coded fuzzy based (dnafz) s-boxes: Application to robust image encryption using hyper chaotic maps," IEEE Access, vol. 9, pp. 14284–14305, 2021.

[61] Han, Xintong, et al. "A new set of hyperchaotic maps based on modulation and coupling." The European Physical Journal Plus 137.4 (2022): 523.

[62] A. Shafique and J. Ahmed, "A color image encryption algorithm based on chaotic map and discrete wavelet transform," in 2022 Global Conference on Wireless and Optical Technologies (GCWOT), pp. 1–5, IEEE, 2022.

[63] S. Zhu and C. Zhu, "Security analysis and improvement of an image encryption cryptosystem based on bit plane extraction and multi chaos," Entropy, vol. 23, no. 5, p. 505, 2021.

[64] A. Anees and Z. Ahmed, "A technique for designing substitution box based on van der pol oscillator," Wireless Personal Communications, vol. 82, pp. 1497–1503, 2015.

[65] A. Anees and Y.-P. P. Chen, "Designing secure substitution boxes based on permutation of symmetric group," Neural Computing and Applications, vol. 32, pp. 7045–7056, 2020.

[66] R. Ratan and A. Yadav, "Security analysis of bit-plane level image encryption schemes.," Defence Science Journal, vol. 71, no. 2, 2021.

[67] A. Anees, I. Hussain, A. Algarni, and M. Aslam, "A robust watermarking scheme for online multimedia copyright protection using new chaotic map," Security and Communication Networks, vol. 2018, 2018.

[68] A. Anees, T. Dillon, S. Wallis, and Y.-P. P. Chen, "Optimization of dayahead and real-time prices for smart home community," International Journal of Electrical Power & Energy Systems, vol. 124, p. 106403, 2021.

[69] A. Anees and Y.-P. P. Chen, "Discriminative binary feature learning and quantization in biometric key generation," Pattern Recognition, vol. 77, pp. 289–305, 2018.

[70] M. Li, S. Pan, W. Meng, W. Guoyong, Z. Ji, and L. Wang, "Medical image encryption algorithm based on hyper-chaotic system and dna coding," Cognitive Computation and Systems, vol. 4, no. 4, pp. 378–390, 2022.

[71] V. Sangavi and P. Thangavel, "An exotic multi-dimensional conceptualization for medical image encryption exerting rossler system and sine map," Journal of Information Security and Applications, vol. 55, p. 102626, 2020.

[72] H. Liang, G. Zhang, W. Hou, P. Huang, B. Liu, and S. Li, "A novel asymmetric hyperchaotic image encryption scheme based on elliptic curve cryptography," Applied Sciences, vol. 11, no. 12, p. 5691, 2021.

[73] A. Bisht, M. Dua, S. Dua, and P. Jaroli, "A color image encryption technique based on bit-level permutation and alternate logistic maps," Journal of Intelligent Systems, vol. 29, no. 1, pp. 1246–1260, 2020.

[74] M. Samiullah, W. Aslam, H. Nazir, M. I. Lali, B. Shahzad, M. R. Mufti, and H. Afzal, "An image encryption scheme based on dna computing and multiple chaotic systems," IEEE Access, vol. 8, pp. 25650–25663, 2020.

[75] I. Hussain, A. Anees, A. H. Alkhaldi, M. Aslam, N. Siddiqui, and R. Ahmed, "Image encryption based on chebyshev chaotic map and s8 s-boxes," Optica Applicata, vol. 49, no. 2, pp. 317–330, 2019.

[76] K. Mali, S. Chakraborty, and M. Roy, "A study on statistical analysis and security evaluation parameters in image encryption," entropy, vol. 34, p. 36, 2015.

[77] F. Ahmed and A. Anees, "Hash-based authentication of digital images in noisy channels," Robust image authentication in the presence of noise, pp. 1–42, 2015.

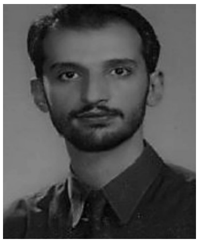
[78] I. Hussain, F. Ahmed, U. M. Khokhar, and A. Anees, "Applied cryptography and noise resistant data security," Security and Communication Networks, vol. 2018, pp. 1–2, 2018.

[79] Sun, Junwei, et al. "Dynamical analysis of HR–FN neuron model coupled by locally active hyperbolic memristor and DNA sequence encryption application." Nonlinear Dynamics 111.4 (2023): 3811-3829.

[80] Sun, Junwei, et al. "Memristor-based neural network circuit with multimode generalization and differentiation on pavlov associative memory." IEEE Transactions on Cybernetics (2022).

ARSLAN SHAFIQUE received PhD degrees in Electrical Engineering from Riphah International University(RIU), Pakistan in 2022. He is currently serving as a Lecturer in the Faculty of Engineering and Applied Sciences at RIU, Pakistan. He has more than fifteen research publications including transactions and high-impact factor journals with an accumulative impact factor of 50+. His research interests include cryptography, secure communication, and machine learning.

MUJEER UR REHMAN received Ph.D. degree (with distinction) from the Faculty of Engineering and Applied Sciences, Riphah International University, Islamabad, Pakistan, in 2022. He is currently serving as Lecturer at School of Science, Technology and Health, York St. John University, York, UK. He is also a Professional Engineer. His research interests include artificial intelligence, non-invasive health care, IoT, cyber security, and multimedia encryption.



KASHIF HESHAM KHAN received a PhD in Industrial Process from Royal Melbourne Institute of Technology in Melbourne, Australia, and completed his B.S. and M.S. degrees (Hons.) in computer science at the COMSATS Institute of Information Technology (CIIT) in 2008 and 2010, respectively. His M.S. thesis concentrated on enhancing grid scheduling approaches for data parallel applications in the context of industrial and sustainable buildings. Currently, he is affiliated with RMIT, Deakin, and Federation University in Australia, and his areas of research interest include cyber security, image processing, and machine learning.

SAJJAD SHAUKAT JAMAL received the Ph.D. degree in mathematics from Quaid-i-Azam University, Islamabad, Pakistan. Currently, he is working as an Assistant Professor with the Department of Mathematics, King Khalid University, Abha, Saudi Arabia. His research interests include mathematics, number theory, cryptography, digital watermarking, and steganography. He has several quality research papers in well-reputed journals on the application of mathematics in multimedia security.

ABID MEHMOOD (MEMBER, IEEE) received his Ph.D. degree in computer science from Deakin University, Australia. He is currently an Assistant Professor at Abu Dhabi University. His research interests include ML, privacy, information security data mining, and cloud computing. He has 12 research publications in well-reputed journals.

SHEHZAD ASHRAF CHAUDHRY received the master's and Ph.D. degrees (with Distinction) from International Islamic University Islamabad, Pakistan, in 2009 and 2016, respectively. He is currently working as an Associate Professor of Cybersecurity Engineering, Abu Dhabi University, Abu Dhabi, UAE. He has authored over 170 scientific publications appeared in different international journals and proceedings, including more than 125 in SCI/E journals. With an H-index of 40 and an I-10 index 91, his work has been cited over 4400 times. He has also supervised over 40 graduate students in their research. His current research interests include lightweight cryptography, elliptic/hyper elliptic curve cryptography, multimedia security, E-payment systems, MANETs, SIP authentication, smart grid security, IP multimedia subsystem, and next generation networks. Dr. Chaudhry was a recipient of the Gold Medal for achieving 4.0/4.0 CGPA in his Masters. Considering his research, Pakistan Council for Science and Technology granted him the Prestigious Research Productivity Award, while affirming him among Top Productive Computer Scientist in Pakistan. For the consecutive three years, he is being listed among Top 2% Computer Scientists across the world in Stanford University's reports. He is also serving as guest editor for many WoS indexed journals and have served/serving as a TPC member of various international conferences. He is also an active reviewer of many WoS indexed journals.

• • •