El-den *et al.*
EURASIP Journal on Advances in Signal Processing (2023) 2023:40
https://doi.org/10.1186/s13634-023-01003-4

EURASIP Journal on Advances in Signal Processing

RESEARCH Open Access

Even symmetric chaotic and skewed maps as a technique in video encryption

B. M. El-den[1], Walid A. Raslan[1] and Ahmed A. Abdullah[1*]

*Correspondence:
ahmed.abdelaleem@deltauniv.edu.eg

[1] Faculty of Engineering, Delta University for Science and Technology, Costal International Road, Mansoura Dakahlia, Egypt

Abstract

The massive growth and use of digital multimedia through computer networks, including video and images, have increased the demand for protecting this digital data. To secure digital video, video encryption is frequently utilized. In this paper, a brand-new video scrambling technique based on two chaotic linearly symmetric maps and one chaotic tent map that has been twisted is suggested. The permutation procedure moves every frame pixel's position using a P-box created by permuting a linearly symmetric chaotic sequence. The diffusion technique employs both linearly symmetric chaos maps and distorted tent maps to create key streams. The keystream closely resembles simple frames because the pixels in the permuted frame indicate which of the two even symmetric chaos maps is replicated each time for the following byte. The information entropy, histogram, neighboring pixel correlation and sensitivity analysis, number of pixels changing regions (NPCR), and unified mean change intensity are used to thoroughly evaluate the recommended method's capacity to improve performance and security (UACI). Comparatively to other methods, the suggested algorithm is resistant to clipping, salt and pepper noise, speckle noise rotation assaults, and clipping. This positive outcome indicates that the plan can be successfully implemented for secure video communication applications.

Keywords: Encryption video, Chaotic map, Even symmetric, Skewed map

1 Introduction

Security of multimedia data has grown in importance in information communication and transmission due to the quick development of computer and Internet technology. For instance, only authorized parties can access multimedia content for video-on-demand, Internet television, video telephony, video conferencing, and military applications. H.264/AVC advanced video coding is a popular format for coded video and one of the most recent video compression standards. It can enhance the video compression efficiency, allowing for greater flexibility in storing and transferring videos when compared to Moving Picture Experts Groups [1].

Before sending, certain private videos need to be secured. One method to accomplish this is encryption. A technique or combination of techniques used to safeguard multimedia content generally provides multimedia security. Concerning H264 video encryption techniques, the issue is how to design a secure and fast encoding system in

which the encryption algorithm is incorporated into the encoding process with minimal additional computational burden while providing acceptable high-level security [1]. Unlike the conventional cryptographic techniques, which are based on discrete mathematics, chaos-based crypt algorithms rely on the complex dynamics of continuous time dynamical systems and also deterministic nonlinear maps. As such, it can provide faster schemes for information security processing with desired data protection, which is crucial for multimedia data transmission over fast communication channels, especially on the broadband Internet [2]. Motivated by this desire and need, in recent years a great deal of effort has been devoted to developing various chaos-based image and video encryption and decryption methodologies and algorithms [3]. Some properties of cryptographic and chaotic maps algorithms are similar, such as sensitivity to changes in initial conditions and parameters, randomness-like behavior, and unstable periodic orbits with long periods. As a result, the chaotic map parameters have served as a key to the encryption algorithm [1].

The following essential characteristics of chaos mapping have led to the development of chaos-based video scrambling systems in recent years: Aperiodicity, random behavior, and sensitivity to beginning conditions are all heavily researched topics. Pixel replacement and diffusion make up a typical chaos-based video encoding process. To break the high correlation of pixels inside the chosen frames, *P-boxes* are often generated, and the positions of the pixels are swapped during the permutation stage. As pixel values are modified consecutively during the diffusion phase, virtually every pixel in the crypto frame is affected by tiny changes in the plain frame. However, numerous permutation-spreading chaos-based video encryption techniques have recently been defeated.

The key, which is the root cause, is the sole factor that influences the keystream utilized in the diffusion stage. If the initial key stays the same, the keystreams generated to encrypt different plaintexts are identical. Therefore, an attacker can gain access to the keystream by both known-plaintext and chosen-plaintext assaults. As a result, the encryption technique becomes a straightforward, already-broken permutation design. To improve security, several researchers suggested connecting the plaintext and the key stream of diffusion. The produced keystream will differ even if the key remains the same and the plaintext stays the same. This is thus because, in a chaotic system, the plaintext is used to control the number of iterations.

This approach is a hit to face up to the known-plaintext attack and chosen-plaintext assault. However, there are a few real issues that persist. First, it is well known that when performed with finite precision, a one-dimensional chaotic map has periodic issues, which can also lead to a subsequent decrease in security. The critical region of a low-dimensional chaotic map is also quite constrained. Second, there will be a significant rise in the number of chaotic machine instances in new releases. Third, the P-container continues to be irrelevant to the plaintext in permutation [4]

2 Literature survey

Multimedia technology security and privacy concerns have grown significantly in importance. Secure communication is necessary for numerous multimedia applications. The amount of protection needed depends on how sensitive the data included in these applications is. Several cryptographic techniques have been developed for

protecting streaming video to solve the issue of processing overhead and satisfy the security demands of real-time video applications with high-quality video compression. The majority of these methods aim to improve the appearance and performance of the encryption process.

The security of the entire *MPEG* stream is ensured by full encryption technology and straightforward algorithms employing industry-standard encryption approaches. There are currently no efficient methods that can defeat encryption systems like triple *DES* and *AES*. This algorithm is not suitable for huge videos because it is quite sluggish, especially when using Triple *DES*. Real-time video scrambling has an unacceptable overhead due to the scrambling operation's addition of lag.

Pure permutation algorithms merely use permutations to scramble the bytes within the frames of a *MPEG* stream. The use of the pure permutation technique should be carefully evaluated because, as the authors of [5] demonstrate, it is susceptible to known-plaintext attacks. This is because by comparing the ciphertext to known frames, an attacker or hacker can quickly determine the secret permutation list. All frames can be quickly decrypted once the permutation list is known or known. It should be noted that the replacement list based on Shannon's theorem can be decoded with just the knowledge of the I-frame of one of the *MPEG* streams.

Instead of mapping *8x8* blocks to 1×64 vectors in a "*zigzag*" order, the zigzag permutation approach [6] employs a random permutation list (secret key) to map a single 8×8 block to a 1×64 vector. The computing complexity of the zigzag order mapping and the random permutation list mapping is equal; therefore, the encryption and decryption procedure only slightly slows down the video compression and decompression process. Attacks using known-plaintext and ciphertext-only techniques can exploit this mechanism.

Techniques for Selective Encryption are discussed to decrease processing overhead [4] and meet the security needs of real-time video applications [7]. Using elements of the MPEG layer structure, this technique seeks to encrypt various layers of a few selected MPEG stream segments (e.g., encryption of all headers and I-frames, encryption of all I-blocks of I-frames, P-frames, and B-frames). Basic selective encryption is based on the MPEG I frame, P Frame, and B Frame structures. Only I-frames are encrypted since P- and B-frames are potentially useless without knowing the accompanying I-frame.

A brand-new video encryption technique called *VEA* was suggested by the author of [8]. The statistical characteristics of the *MPEG* video standard and the Symmetric Key Algorithm standard are used by video encryption algorithms to lessen the amount of encrypted data. *Algorithm I, Algorithm II (VEA), and Algorithm III* are four alternative video encryption techniques that a researcher introduced in [9] (*MVEA*). The first technique employs Huffman codeword permutations in *I-frames*. This technique performs both compression and encryption in one step. The algorithm's secret ingredient is the permutation p. The regular *MPEG* Huffman codeword list is frequently replaced with this confidential portion. Both known plaintext and ciphertext-only attacks can use this technique. Algorithm II (*VEA*): I frames carry the most crucial information about *MPEG* video. Therefore, we just need to *XOR* them with an m-bit binary key [10–14] to encrypt the desired sign bits of a block of *I-frames' DC* coefficients. Complete security is offered by this encryption. However, this is not feasible for mass media applications like

video-on-demand services and other comparable applications. But when the key size is very little, the entire method is condensed and referred to as a Vinegary-like cipher.

The algorithm proposed by Arif et al. [15] is based on logistic maps. The proposed algorithm uses the plaintext image to generate a hash, which is then divided into four parts, each of which is used as an initial parameter input for the logistic maps to generate four pseudorandom number arrays. The algorithm then performs row and column permutations using the first and second keys, respectively. An XOR operation is performed on the resulting image using the third key. The last step is to perform a substitution on the image using either AES S-Box or AES inverse S-Box based on the fourth generated key. However, the algorithm has a large key space and is short time-consuming.

Alawi et al. [16] proposed an alternative video encryption algorithm, in which the key is generated pseudo-randomly by the ChaCha algorithm. The H.264/A VC encoding algorithm encodes the video into multiple slices; the key semantic elements in the slice can be selectively encrypted; the key is generated by a pseudo-random generator and updated in real-time [17]. However, in the existing scheme, although the video security problem can be solved well, the key is pseudo-randomly generated according to the traditional key negotiation process, and the security mainly depends on some number theoretic problems with computational complexity, such as the discrete logarithm problem, integer factorization problem, and elliptic curve discrete logarithm problem.

The authors in [18] propose a selective encryption scheme using singular value decomposition and chaotic systems in order to overcome such issues. The proposed method ensures the confidentiality of video streams originating from devices that have minimal resources and that are mostly used in a smart-traffic management system. The National Infrastructures (NIS) directive has identified several critical sectors including transport and the proposed method could be used as an efficient tool to secure the information that is created and transmitted through a smart traffic system.

Two methods of the Chaos-based encryption methodology are presented in [1] for steganography and cryptography processes of different I-frames with different resolutions of compressed videos sequences H.264/AVC. In [19], A suggested a method that uses chaotic maps to shuffle pixels inside a frame. The algorithm belongs to the compression-encryption algorithm class. The pseudo-random generator is designed using two chaotic maps Hitzl-Zele map and Tinkerbell map in [20]. According to two chaotic maps, a secure pseudo-random number generator was designed, and a binary sequence was generated by this pseudo-random number generator [20]. No attacks techniques have been checked.

Hui Xu [21] suggested a robust video encryption scheme based on an H.246 compressed code stream and a cross-coupling chaotic system for keystream generation. The proposed algorithm had lower time overhead and did not cause a significant increase in bit rate. Dua [22]proposes a fast and secure method of video encryption using 3D an Intertwining Logistic Map (ILM) with cosine transformation to generate a complex chaotic sequence. The keys produced by combining SHA-2 with cosine-based ILM are more uniform, and nonlinear.

Maolood et al. [23] suggested a lightweight stream cipher method. Then, it was tested for numerous video samples to check its suitability and authentication in encryption and decryption procedures. After testing many characteristics such as

differential analysis, correlation analysis, information entropy and histogram analysis, their method showed a higher security and lower calculation time compared with state-of-the-art encoding methods [23].

The security of current microgrid communication is guaranteed by cryptographic systems, such as the Advanced Encryption Standard (AES) [24]. In AES and other symmetric key cryptographic systems, a communicating party uses a key to encrypt data messages, and the other party uses the same key to decrypt data messages. Those keys, which are preshared by two remote parties, have to be generated and distributed securely

The application of pseudo-random number generators for IoT is increasingly being studied [25]. Unsub Zia et al. [25] have been proposed generalized symmetric maps with a user-chosen chaotic map by changing the adaptive control parameter used to generate a pseudo-random key, this model was tested on raspberry pi 3b+ and raspberry pi zero. However, chaotic cryptography deals not only with stream ciphers, but also pseudo-random numbers generators (PRNGs) [25]

Recently, a productive chaotic pseudo-random number generator was created to produce keystreams for encrypting the syntactic parts of *H.264/AVC* video. From the standpoint of effectiveness and security, the signs of the intra-prediction mode (*IPM*), trailing ones (*T1s*), non-zero coefficients (*NZ*), and motion vector difference (*MVD*) are selected for selective encryption. The proposed plan effectively safeguards the video's commercial worth. According to experimental findings, his *H.264/AVC* maintains the exact same bitrate and has very little overhead, and therefore, the encryption history has no impact on the coding efficiency of the format.

The novel video encryption method provided in this paper is based on various chaos maps. The linearly symmetric chaotic map's iteration time is not fixed during the permutation process to prevent transition effects, yet the *P-box* changes even though the initial values are kept constant because it is connected to the plaintext. The keystream is produced via the diffusion process using two even-symmetric chaotic maps. Instead of lengthening the iteration time, a linearly symmetric chaos map is created using the pixels in the reordered selected *I-frames.* This chaos map iterates for each subsequent byte in the keystream. On the other hand, employing numerous chaotic systems practically solves the issues of short periods and tiny key spaces in one-dimensional chaotic maps. Given that it was thought to have acceptable statistical features, linear symmetric chaos mapping was chosen [4]. In the plan, a curved tent map is also employed. The approach can be applied to other fields, such as text encoding because it treats the chosen frames as vectors.

The main contribution of this paper can be summarized as follows: -

- The novel video encryption method provided in this paper is based on various chaos maps.
- Employing numerous chaotic systems practically solve the issues of short periods and tiny key spaces in one-dimensional chaotic maps
- Selective encryption encrypts part of a compressed data file while sending the rest unencrypted. Even a small number of encrypted bits can cause more file damage

with this method. In contrast to the full file being encrypted bit by bit, only the sensitive parts are altered.

- The proposed video encryption scheme will be tested by different video files.
- The proposed scheme is compared with the state-of-the-art video encryption schemes. Results indicate the proposed encryption scheme outperforms other video encryption methods given in the literature and provides higher resistance against attacks while requiring less computational complexity.
- The approach can be applied to other fields, such as text encoding, because it treats the chosen frames as vectors.

The remaining paper is arranged as follows. The related work is explained in sections III, respectively. The proposed video encryption algorithms are described in Section IV. An evaluation of the suggested method's performance is presented in Section VI. The effects of various attacks, experimental assessments, and security analyses are covered in Section VII. The proposed rule's conclusions are presented in Section IX.

3 Related work

3.1 Construction of even symmetric chaotic map

As described in [26], even-symmetric chaotic maps (*ESCMs*) have been proven with excellent statistical properties. Construct a linearly symmetric chaotic system using the following three steps:

Step1: Construct an even-symmetric map $T: [0, 1] \rightarrow [0, 1]$ such that for any real number $z \in [0, 1]$, $T(1 - z) = T(z)$ and the unique invariant measure density $f^{*}T(x) = 1$. Here, a piecewise-linear map T is constructed as Eq. (1):

$$T(z) = \begin{cases} \frac{z}{d} z \in [0, d] \\ \frac{(z-d)}{(0.5-d)} z \in [d, 0.5] \\ \frac{(1-z-d)}{(o.5-d)} z \in [0.5, 1-d] \\ \frac{(1-z)}{d} z \in [1-d, 1] \end{cases} \tag{1}$$

Step2: (x) is the desired map:

$$q(x) = \int_0^x 3(2x-1)^2 .dx = 4x^3 - 6x^2 - 3x \tag{2}$$

$$q^{-1}(x) = 0.5 + 0.5(2x-1)^{1/3} \tag{3}$$

$$F(x) = q^{-1}\{T[q(x)]\}, x \in (0, 1) \tag{4}$$

Which stratifies $f_F^*(x) = 3(2x-1)^2, q(x) = \int_0^x f_F^*(x).dx$. so it uses to generate real value sequences where $x_0 = 0.5$ should be avoided as the initial value.

Step 3: generation of chaotic *BB* sequences, a binary function *H(x)* as Eq.(5) is used:

$$H(F(x)) = \begin{cases} 0, \textit{iff}\,(x) \in (0, 0.378) \cup (0.5, 0.622) \\ 1, \textit{iff}\,(x) \in (0.378, 0.5) \cup (o.622, 1) \end{cases} \tag{5}$$

The binary function *H (F(x))* satisfies the symmetry condition in Eq.6.

$$H(a + e - x) = 1 - H(x), x \in [a, e] \tag{6}$$

A. ***The Proposed method of Video Encryption technique:*** Permutation operator using *ESCM 320 × 560* Gy-scale frames of estimate $M \times N$ is displayed by a one-dimension vector $P = \{P_1, P_2, ..., P_{M \times N}\}$. The *I-* frames can be permutated by the taking after steps:

B. Set a real-value parameter $y_1 \in (0, 1)$.

C. Get the introductory condition of *ESCM*

$$x_0 = \frac{(p_j + 1)xy_1}{N} \tag{7}$$

D. Here *J* represent the number of parameter chosen by the user.

E. Iterate $x_{i+1} = (x_i)$ for *L* times to avoid the transient effect where $L = 200$.

Continue to iterate $x_{i+1} = F(x_i)$ for $M \times N$ times and get a real- value sequence $x = \{x_1, x_2, x_{3,}..._{,\times N}\}$.

- Keep the position of *xi* unchanged and sort the sequence $x = \{x_1, x_2, x_3, ..., x_{i-1}, x_{i+1}, ..., x_{M \times N}\}$ in ascending order to obtain a new sequence $x = \overset{'}{x}_1$.
- Find the positions of $X = \{\overset{'}{x}_1, \overset{'}{x}_2, \overset{'}{x}_3, \ldots, \overset{'}{x}_i, \overset{'}{x}_{i-1}, \overset{'}{x}_{i+1}, \ldots, \overset{'}{x}_{M \times N}\}$ in *X* and denote them as $T = \{t_1, t_{2...}\, t_{M \times N}\}$ where $x_{ti} = \overset{'}{x}_1$ and $x_{tj} = \overset{'}{x}_j = x_j$.
- Shuffle $P = \{P_1, P_2, ..., P_{M \times N}\}$ by using *T* as the P- box and get $\overset{'}{p} = \left\{\overset{'}{p}_1, \overset{'}{p}_2, \ldots, \overset{'}{p}_{M \times N}\right\}$ such as $\overset{'}{P}_i = P_{ti}$ and $\overset{'}{P}_j = P_{j.}$

4 The proposed method of video encryption technique

Selecting the Encryption frames: A method known as selective encryption encrypts a portion of a compressed data file while sending the other portion unencrypted. With this tactic, even a tiny number of encrypted bits can cause more file harm. Only the sensitive parts are altered, as opposed to the full file being encrypted bit by bit [27]. Furthermore, selective encryption requires less overall encryption work, which conserves system resources. For instance, only a portion of the video stream is encrypted, which explains this. The *H.264 I-frame, P-frame, and B-frame* structures serve as the foundation for basic selective encryption. As shown in Fig. 1 [28], encrypting only *I-frames* is theoretically pointless if *P- and B-frames* are unaware of the accompanying *I-frames*.

While maintaining the security of the original file, the suggested method selectively encrypts a section of a compressed video file. The system becomes more complicated

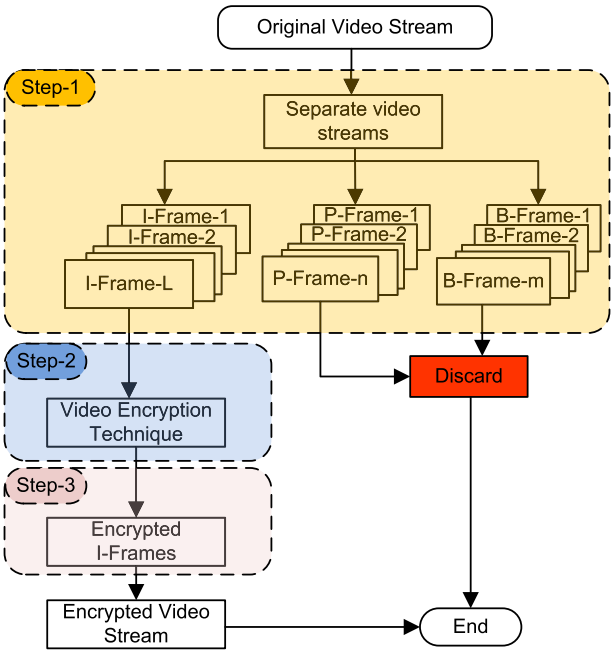


Fig. 1 Framework of video encryption stream.

even if it takes less time to encode the video file. The concept behind this approach is to encrypt different levels of chosen *H.264* stream segments using the *H.264* hierarchy's features.

- Step 1: Separating Video into Frames.
- Step 2: Applying the proposed Video Encryption Technique.
- Step 3: Collecting Encrypted frames to Encrypted Video Stream.

The video stream can be divided into frames in step 1. Three different frame kinds exist the *I, B, and P frames.* The suggested approach chooses just the *I-frames* to move on to the next stage and discards the B-frames and P-frames because they can be derived from the *I-frames*, and encrypts only the I-frames. Fig. 2 displays a flowchart outlining how the video scrambling method operates.

5 Frame encryption & decryption technique

In this approach, the keystream and the reordered image are encrypted by two even symmetric chaotic systems with different control parameters. In this technique as shown in Fig. 3, the two *ESCMs* with distinct parameters (x_0,) and ($x`_0$, $d`$) are simply referred to as the symmetric chaotic maps ((x)) and $H`(F`(x))$, respectively. ($y2$, $y3$) There are two detailed diffusion algorithms.

A. ***The First Round***

B. Using the permutation method in the previous to get the shuffled image $P` = \{p_1`, p_2`, \ldots, p`_{M\times N}\}$.

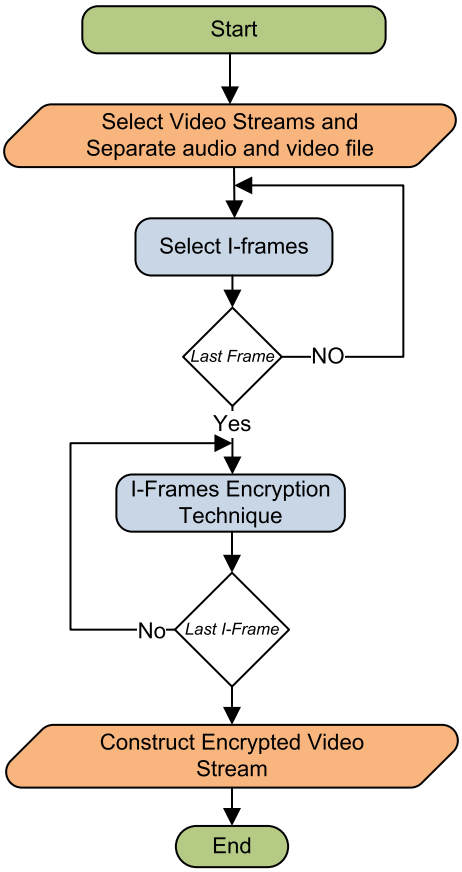


Fig. 2 Flowchart of video encryption technique stream.

C. Calculate $x_0 = \frac{(p_{j+1}).x.y_2}{N}$ and $x`_0 = \frac{(p_{j+1}).x.y_3}{N}$.
D. Iterate $H(f(x))$ with the initial value x_0 for 8 times and obtain 8 bits denoted as b_1.
E. Calculate the ciphered pixel value D_i by using the currently operated pixel $P`_i$, the previous pixel of permuted image $p`_{i-1}$ and bi ($i=1, 2, ..., M \times N$).

$$D_i = \begin{cases} D_{i-1} \oplus p`_i \oplus mod\left(\text{floor}\left(y_3.x.2^{48}\right), 2^8\right), i = j \\ p`_i \oplus mod\left(p`_{i-1} + b_i, 2^8\right), i \neq j \end{cases} \tag{8}$$

Here $\oplus$ is bitwise *XOR* operator and set $p`_o = p_j$. So, the inverse formula of the previous Equation for p_i' can be described as:

$$p`_i = \begin{cases} D_{i-1} \oplus D_i \oplus mod\left(\text{floor}\left(y_3.x.2^{48}\right), 2^8\right), i = j \\ D_i \oplus mod\left(p`_{i-1} + b_i, 2^8\right), i \neq j \end{cases} \tag{9}$$

- Calculate $v_i = D_i \ mod\ 2$ and choose which of two even symmetric chaotic systems will be iterated to create the next component b_{i+1} within the keystream: i. If $v_i = 0$, iterate $((x))$ for 8 times to obtain b_{i+1}; ii. If $v_i = 1$, iterate $H'(F'(x))$ for 8 times to obtain b_{i+1}.
- Let $i = i + 1$, and return to D_i equation.
- Get the results of the first encryption round $D = \{D_1, D_2, ..., D_{M \times N}\}$.

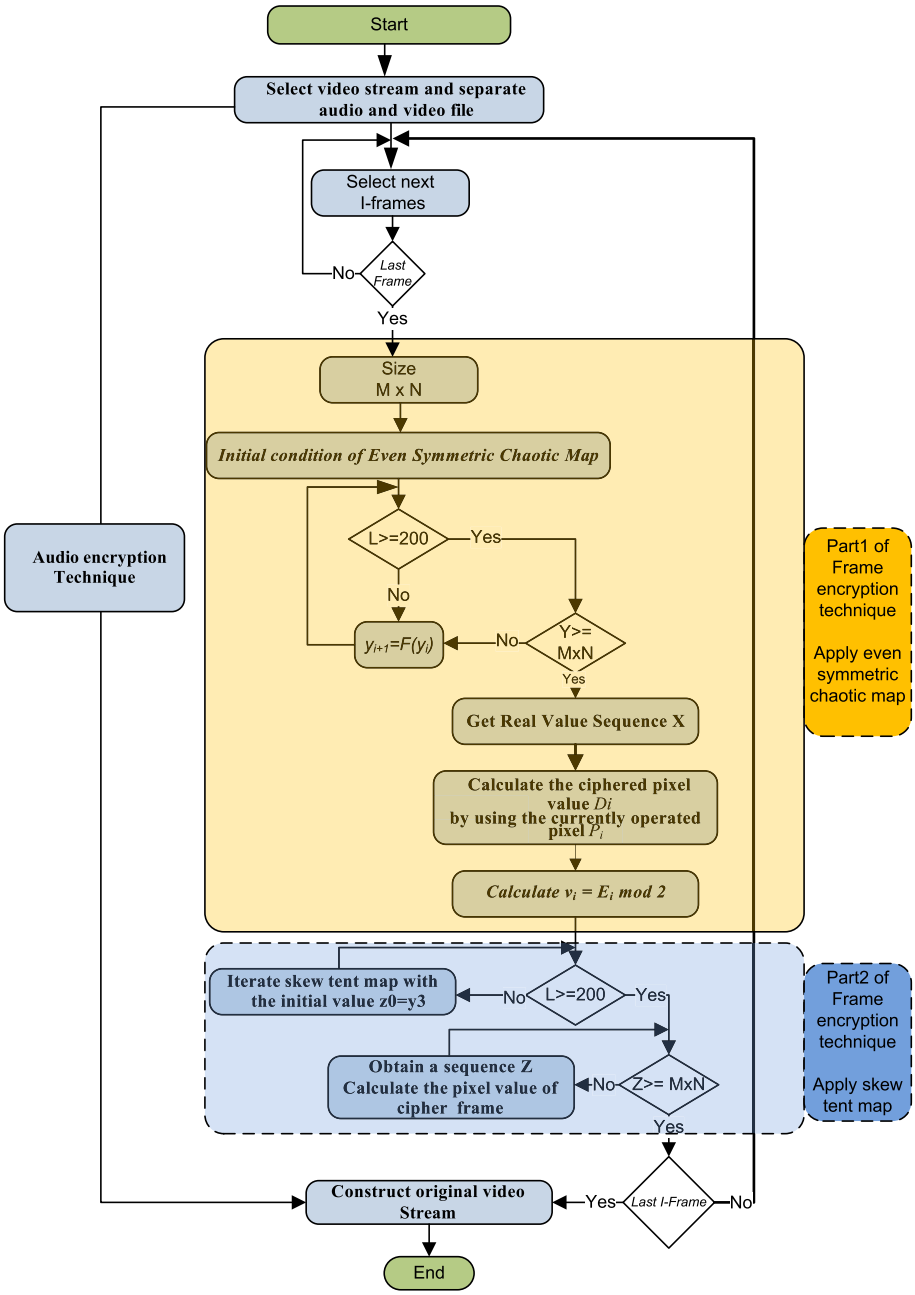


Fig. 3 Flowchart of video encryption using selective even symmetric and skew tent chaotic maps

- ***The Second Round***
- To make strides the security, a skew tent map is additionally utilized within the second round
- Iterate skew tent map as depicted within the next condition with the introductory value $z_0 = y_3$for L times to induce freed of temporal effect:

$$G(z) = \begin{cases} \frac{z}{q} z \in [0, q] \\ \frac{(1-z)}{(1-q)} z \in [q, 1] \end{cases} \tag{10}$$

- Continue to iterate skew tent map for $M \times N$ times to obtain a sequence $Z = \{z_1, z_2, \ldots, z_{M\times N}\}$ Then calculate the pixel value of cipher image:

$$\emptyset_i = floor(560 \times z_i) \tag{11}$$

$$c_i = c_{i-1} \oplus \emptyset_i \oplus \mathrm{mod}(D_i + \emptyset_i, 560) \tag{12}$$

- Here c_0 is a given constant and $\{c_1, c_2, \ldots, c_{M\times N}\}$ is the final cipher vector. The inverse formula can be described as following:

$$D_i = \begin{cases} c_i \oplus c_{i-1} \oplus \emptyset_i - \emptyset_i if\, \alpha \geq \emptyset_i \\ 560 + (c_i \oplus c_{i-1} \oplus \emptyset_i - \emptyset_i), if\, \alpha < \emptyset_i \end{cases} \tag{13}$$

- ***Frames Decryption***

The decryption procedure includes four steps.

- Iterate Eq. 10 and Eq. 11 to get $\{\emptyset_1, \emptyset_2, \ldots, \emptyset_{M\times N}\}$ and calculate $D = \{D_1, D_2, \ldots, D_{M\times N}\}$ by Eq. 13
- Use Pi' Eq. to obtain $P' = p`_1, p`_2, .., p`_{M\times N}$
- Calculate the initial condition x_0 by $P`_j$ which equals to p_j a regenerate the *P-box T*.
- Remove the effect of permutation from $P`$ by performing the reverse operation of permutation with the *P-box T*. Finally, the plaintext P is recovered.
- In this encryption process, the key stream is obviously related to the plain *I-frames*. The framework of decryption procedure as shown in Fig. 4.

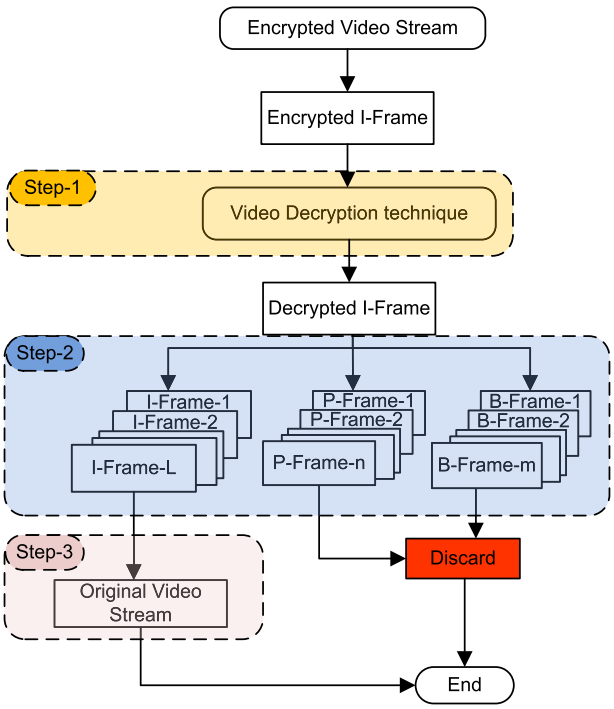


Fig. 4 Framework of video decryption procedure.

6 Results & discussion

Histograms, information entropy, correlation, and sensitivity analysis of nearby pixels (horizontal, vertical, and diagonal) from basic and cryptographic frames, *UACI*, as well as *NPCR*, are all used to evaluate the performance and security of the suggested method. Compared to previous methods, the suggested algorithm is resistant to attacks from clipping, salt-and-pepper noise, and speckle noise rotation.

Statistical attacks should be resistant against a solid permutation and diffusion-based video encryption method. Through some statistical analysis, the performance of the suggested video encryption algorithm is carefully assessed in this section.

6.1 Encryption results

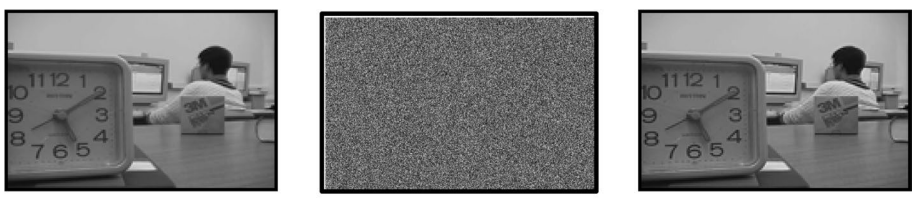
The computer used in the practical test had several features, including an 16.0 GB RAM, an Intel(R) Core(TM) i7-10750H CPU @ 2.60 GHz 2.59 GHz, and an operating system of Windows 10 Pro. Video stream features are video type *H.264*, length of Video is *5.6* Seconds, Num. of Frames *166* frames, Num. of frames per second *30fps*, frame $M \times N$ *is* 560×320 *px*, Num. of *I-frames are 42 I-frame*. The one sample of the selected plain frame and the encrypted frame using the proposed algorithm as shown in Fig. 5. Here, parameters are $(y_1, y_2, y_3) = (0.9487, 0.5192, 0.7538)$ and $(d, d_0, q) = (0.27, 0.13, 0.57)$.

6.2 Histogram analysis

By showing the number of pixels at each grayscale level, the histogram of an image shows how pixels are distributed. A good encryption technique should conceal the plain image's spreading character and prevent information leakage. Therefore, a cipher image's ideal histogram should have a uniform distribution. The suggested algorithm's histograms for the plain picture and encrypted image are contrasted in Fig. 6. The outcome shows that the encrypted image's histogram has a relatively uniform distribution.

6.3 Correlation of adjacent pixels

Original frames with significant visual content consistently have a high correlation between adjacent pixels. To produce scrambled frames with a high degree of correlation between neighboring pixels and be more resistant to statistical attacks, a trustworthy video scrambling algorithm should therefore avoid this problem. Figure 7 displays the distribution of nearby pixels for plain and scrambled frames. It has been demonstrated that the high connection between adjacent pixels in a straightforward frame drastically decreases after scrambling. Next, between the plain image and the cryptographic frame,



(a) (b) (c)

Fig. 5 **a** The original frame (**b**) the encrypted frame (**c**) the decrypted frame

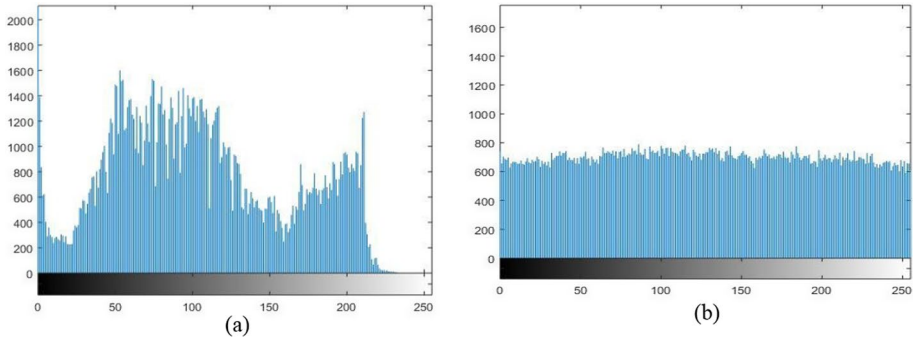


Fig. 6 **a** Histogram of original FRAME, (**b**) histogram of Encrypted frame

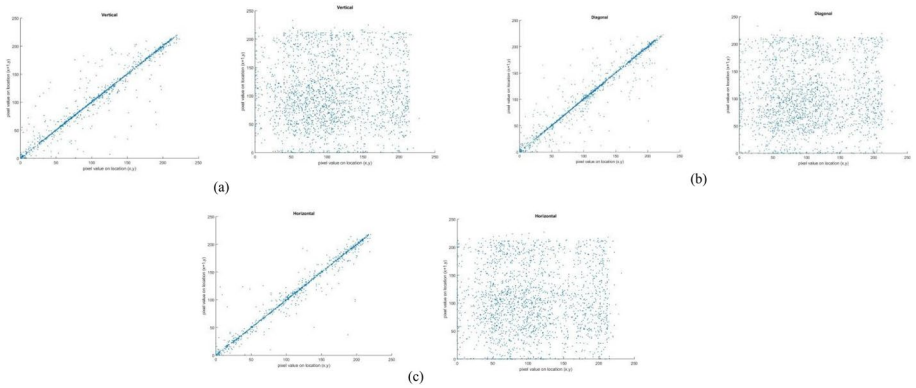


Fig. 7 **a–c** Correlation of adjacent pixels in (vertical, horizontal, diagonal) from plain & cipher frames

randomly choose 1,000 pairings of two adjacent pixels (horizontal, vertical, diagonal), and then calculate the coefficient for each pair according to Fig. 7.

$$Cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))\left(y_i - E(y)\right) \tag{14}$$

$$r_{xy} = \frac{Cov(x,y)}{\sqrt{D(x)}.\sqrt{D(y)}} \tag{15}$$

Here, x and y denote gray-scale values of two adjacent pixels.

6.4 Information entropy

In information theory, the most significant aspect of randomness and irrationality is information entropy. Use Eq. 16 to determine the entropy *H(s)* of the message source *s*.

$$H(s) = \sum_{i=0}^{2^N-1} p(s_i) log_2 \frac{1}{p(s_i)} \tag{16}$$

where (*si*) is the likelihood that *si* will appear in message *s*. The ideal entropy value is *8*. For the three encrypted photos using the suggested method in Table 1, the computed information entropy is extremely near to *8*. This indicates that there is little information loss during encryption and that the encryption scheme is safe from entropy attacks.

Table 2 illustrates that the information entropy of the proposed algorithm based on even symmetric chaotic and skewed maps compared to Hitzl-Zele and Tinkerbell maps [20], Chaotic maps [19], 4-D hyperchaotic [17], RSVE [29] Arnold Algorithm, and Scramble chaotic. All the entropy outcomes of the proposed algorithm are contrasted with those from alternative encryption techniques. So, the proposed scheme is immune and resistant against entropy attacks.

6.5 Sensitivity analysis

A good video coding technique must be sensitive to both key and plain images to withstand differential and brute-force attacks. Two of the most common sensitivity tests, *NPCR* (Number of Pixel Change Range) and *UACI* (Unified Average Changing Intensity), look at changes in the area between two frames. If $c1(i,j)$ and $c2(i,j)$ denote the two images $1 \le i \le M, 1 \le i \le N$, respectively, then the determination of *NPCR* and *UACI* by using the following equations [31].

$$D_i = \begin{cases} 0, if c_1(i,j) = c_2(i,j) \\ 1, if c_1(i,j) \neq c_2(i,j) \end{cases} \tag{17}$$

$$NPCR = \frac{\sum D(i,j)}{M \times N} \times 100\% \tag{18}$$

$$UACI = 255 \times \frac{\sum |c_1(i,j) - c_2(i,j)|}{M \times N} \times 100\% \tag{19}$$

6.6 Key sensitivity

This feature allows us to observe the sensitivity of the cryptosystem's keys. With a slightly altered key, a completely different I cipher frame is produced. After encrypting plain frames, the encrypted frames are decrypted using a slightly different key. The decoding outcome is displayed in Fig. 5. The cipher frame cannot be correctly decrypted without the right key. Table 3 compares both NPCR and UACI results using the suggested algorithm with some existing works. More than *99%* of the pixels in the encrypted image alter their gray value when the key changes significantly, according to *NPCR* and *UACI* computations across cryptographic frames with slightly different keys in Table 3. Because of this, the suggested approach offers excellent key sensitivity.

6.7 Plaintext sensitivity

Small changes in plaintext can lead to large changes in the cryptographic framework called plaintext confidentiality. To thwart differential attacks, an effective cryptographic technique must be sensitive to the plaintext. Randomly select one pixel from each of the three video frames (Video 1, 2, and 3) and change its value slightly. Two rounds of encryption are performed in a simple frame to improve performance. Next, determine

the *NPCR* and *UACI* for each pair of crypto frames. According to Table 3, after two rounds of encryption, a single pixel difference in a simple frame can result in a pixel change of over *99%* of the encrypted frame as shown in Tables 4, 5.

7 Noise and attacks

- ***Chosen-plaintext attacks:*** It is acceptable for a plaintext attacker to have access to a collection of plaintexts and use attacks like rotation and cropping. Noise could be a sporadic type of video. It might be visible in the video as the effects of fundamental physics, such as the photon nature of light or thermal energy inside the picture sensors. It could create media when recording or transmitting a video. Because of the noise, the pixels inside the frames look at a range of intensities rather than as their actual values. Calculating noise ejection is a technique for removing or reducing noise from frames. The noise evacuation algorithms smooth the entire frame and clear out ranges close to contrast limits to reduce or evacuate the perceptibility of noise. Consequently, it can identify specific types of noise and use a variety of techniques to eliminate it. There are several types of image noise, including motivation noise (also known as salt-and-pepper noise), amplifier noise (also known as Gaussian noise), shot noise, quantization noise (also known as uniform noise), film grain, on-isotropic noise, multiplicative noise (also known as sparkle noise), and periodic noise.
- ***Effect of salt and pepper noise:*** This kind of noise is often referred to as impulse noise [32]. Other synonyms include independent noise, random noise, and spike noise. As a result of this noise, which is also known as salt and pepper noise, black and white dots emerge in the frame [33]. Sharp and rapid fluctuations in the frame signal cause this noise to appear in the frame. By using a median filter, the impact of salt and pepper noise is reduced. Before and after the median filter, calculate the peak signal-to-noise ratio (*PSNR*), signal-to-noise ratio (*SNR*), mean square error (*MSE*), and correlation coefficient of the frame with salt and pepper noise as shown in Table 6. The proposed scheme performs better compared with the existing chaotic maps in [1, 19].
- ***Effect of Gaussian noise:*** The nature of this noise model is additive [34] and it has a Gaussian distribution. In other words, the genuine pixel value and a random, Gaussian-distributed noise value are added together to form each pixel in the noisy image. The intensity of the pixel value at each place has no bearing on the noise. The mean and standard deviation, respectively, are the Gaussian distribution noise in the frame. Table 7, "Correlation Coefficient of Frame with Gaussian Noise Before and After Gaussian Filter," calculates the *(PSNR), (SNR),* and *(MSE).*

- ***Effect of speckle noise:*** In frames, speckle noise is both random and deterministic. Video frames are negatively affected by speckle. Due to interference from both constructive and destructive sources, the frames of a generally uniform object with several scattering sources within a resolution cell will have pixel values that change arbitrarily with the position. The main cause of certain frames becoming

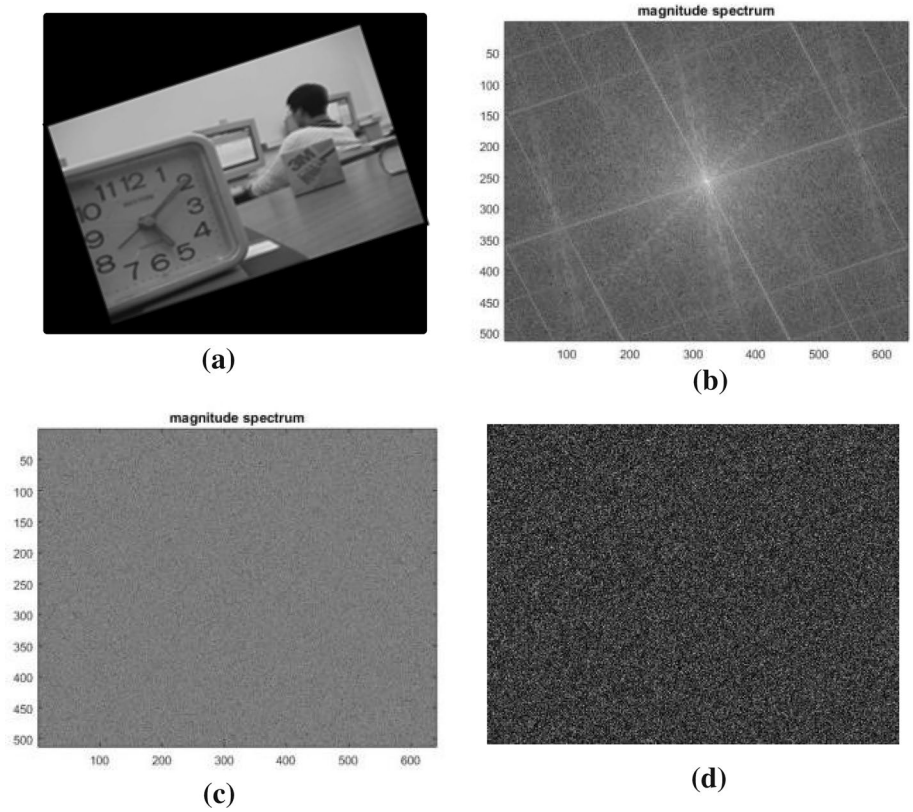


Fig. 8 **A** The rotational attack of original I-frame, **b** The magnitude spectrum of original I-frame. **C** The magnitude spectrum of encrypted rotational I-frame **d** Encrypted the rotational I-frame original I-frame

Table 1 The information entropy by the proposed method in different videos

Video	The information entropy by the proposed method	
	Original frame	Encrypted frame
Video (1)	7.6153	7.9978
Video (2)	7.4271	7.9950
Video (3)	7.2043	7.9913

Table 2 The information entropy by the proposed method in different videos

Algorithm	MVEA [30]	Hitzl-Zele map and Tinkerbell [20]	Chaotic maps [19]	4-D hyperchaotic [17]	RSVE [29]	Arnold Algorithm	Scramble chaotic	Proposed Algorithm
Entropy	7.9673	7.5993386	7.2785	7.9899	7.6519	7.9271	7.9967	7.9978

faulty is speckle noise. It is granularly patterned multiplicative noise. Speckle noise has the characteristics of being a multiplicative noise that is directly proportional to the local gray level in any given location. In terms of statistics, the signal and the noise are independent. A single pixel's sample mean and variance are the same

Table 3 The number of pixels changes range and unified average changing intensity

Algorithm	Number of pixels change range and unified average changing intensity	
	NPCR %	UACI %
Proposed Algorithm	99.60	33.462
Hitzl-Zele map and Tinkerbell [20]	99.60	–
Chaotic maps [19]	99.803	35.4426
ChaCha20 [23]	99.3	33.71

Table 4 Number of pixels change range and unified average changing intensity for i-frames with different keys

Different keys	Number of pixels change range and unified average changing intensity	
	NPCR %	UNCI %
$D * 10^{-10}$	99.55	33.55
$Y1 * 10^{-10}$	99.58	33.68
$Q * 10^{-10}$	99.23	33.74

Table 5 Number of pixels change range and unified average changing intensity for I-frames with different Keys

Different videos	Number of pixels change range and unified average changing intensity	
	NPCR %	UACI %
Video (1)	99.60	33.46
Video (2)	99.42	33.55
Video (3)	99.55	33.47

Table 6 Salt and pepper noise

Parameters	Proposed algorithm		Logistic chaotic map [1]	Chaotic maps [19]
	Before median filter	Apply median filter		
PSNR	28.3400	39.1679	11.96	33.58
MSE	95.2969	9.9150	4.14×10^3	–
Correlation Coefficient	0.9857	0.9985	0.0932	–

as the mean and variance of the surrounding area, [31]. Table 8 computes correlation, *MSE*, *PSNR*, and *SNR*. Speckle coefficient for a frame with Speckle noise before and after median filter [30, 35].

Table 7 Gaussian noise with Gaussian filter

Parameters	Gaussian noise	
	Proposed algorithm	Apply Gaussian filter
PSNR	20.2051	28.5486
MSE	620.2537	90.8276
Correlation Coefficient	0.9155	0.9867

Table 8 Speckle noise

Parameters	Proposed algorithm	
	Before median filter	Apply median Filter
PSNR	35.3400	37.1327
MSE	55.2969	31.6092
Correlation Coefficient	0.9894	0.9952

Table 9 Features of encryption and decryption frames and elapsed time for traditional techniques and proposed technique

Parameter	MVEA [30]	Arnold algorithm	Scramble chaotic	Hitzl-Zele map and Tinkerbell [20]	Proposed algorithm
No. of frame	166	166	166	166	166
Size of frame	320px × 560px	320px × 560px	320px × 560px	320px × 560px	320px × 560px
Encrypt time (Sec)	28.93512	20.1459	22.1456	197	18.1513
Decrypt time (Sec)	25.17515	49.8718	19.1106		20.7021

- ***Rotational attacks:*** Appropriate selection weights and learning rates considerably limit the impact of rotation on video frames [36, 37]. The experimental findings demonstrate the algorithm's resistance to rotational attack as shown in Fig. 8.
- ***Channel attacks:*** After the proposed technique was successfully implemented, this research carried out an attack on the encrypted file such that, upon decryption, the user received the file in a format that was incomprehensible to humans, allowing hackers to carry out attacks on encrypted frames to compromise data. An example of channel attacks is the effects of Paper and Salt before the receiver can view the video after channel transmission receives the video, the video can be subjected to background sounds like salt and pepper creaking. Show the results of the parameters before in Tables 9, 10 the filters. Table 11 displays the parameters' results after the application median filter [37, 38].

Table 10 Density before applying a median filter

Parameter	Attacks	Density before the apply median filter			
		MVEA [30]	Arnold algorithm	Scramble chaotic	Proposed algorithm
PSNR	Salt & pepper	26.7321	27.7735	28.300	28.3400
MSE		100.529	108.5744	95. 969	95.2969
Correlation Coefficient		0.9895	0.9836	0.9850	0.9857
PSNR	Speckle noise	27.6897	29.6668	27.3741	35.3400
MSE		75.7536	70.2106	101.1936	55.2969
Correlation Coefficient		0.9796	0.9836	0.9864	0.9894

Table 11 Density after applying a median filter

Parameter	Attacks	Density after applying a median filter			
		MVEA [30]	Arnold algorithm	Scramble chaotic	Proposed algorithm
PSNR	Salt & pepper	39.6792	34.6957	38.1923	39.1679
MSE		9.9550	22.0549	9.9160	9.9150
Correlation Coefficient		0.9975	0.9966	0.9965	0.9985
PSNR	Speckle noise	34.7231	31.7462	32.1345	37.1327
MSE		33.4945	43.4969	40.4322	31.6092
Correlation Coefficient		0.9943	0.9933	0.9941	0.9952

8 Comparison of proposed technique with existing algorithms

8.1 Encryption time for the proposed algorithm & Existing Algorithms

The performance of the suggested technique will now be compared to the most widely used techniques already in use. The time analysis of several algorithms is quantified in the table that is provided below. Using the *MVEA*, Arnold Algorithm, and Scramble chaotic and suggested technique, the same video *H. 264* file that was previously utilized is encrypted in this case. Table 9 shows the characteristics of the encryption and decryption frames as well as the amount of time taken for the previously used and newly proposed methods.

8.2 Compare the effect of noises

The effectiveness of the suggested approach against noises and attacks will be evaluated against the most well-liked existing techniques. After the noise has been disclosed, Table 10 will quantify the *PSNR*, *MSE*, and correlation coefficient analyses of several techniques.

8.3 Compare the effects of noises after applying the median filter

The performance of the suggested strategy and the most well-liked existing techniques can be improved by applying a median filter. The performance of *PSNR*, *MSE*, and correlation coefficient is shown in Table 11. When it is greater than *40*, *PSNR* is good. When the value is zero, *MSE* has an excellent value. When the correlation coefficient is more than one, it has a good value. The *PSNR*, *MSE*, and correlation coefficient values of the suggested algorithm are the best.

8.4 Computational and complexity analysis

The computational and complexity analysis has been discussed in this section. One of the major challenges in applying real-time cryptographic schemes to various video types is the encryption and decryption of entire video frames. The proposed scheme in this paper contributes to mitigating this problem by reducing the computational complexity of the encryption scheme based on selecting the encryption frame. Only the sensitive parts will be encrypted. This technique resulted in a low computational complexity of the cryptographic scheme. Finally, the proposed encryption scheme reduces encryption computational complexity while increasing encryption speed. The complexity of the proposed algorithm is defined by the computations and iterations of the encryption/decryption calculations. Considering the linear computation of every iteration, for pixel encryption and decryption of every frame, the total complexity of every frame is $\Theta(n^2)$ meaning the proposed algorithm depends on the rows and columns of every frame (frame width and frame height) and also it depends on the number of frames as shown in Table 9.

9 Conclusions

This research suggested a novel image encryption method based on compound chaotic maps. The technique uses a skew tent map, two even symmetric chaotic maps, and a permutation-diffusion architecture to shuffle and disperse the pixels in a plain image. The scheme's key space is sufficiently wide to fend off brute-force assaults, and the cipher frames are shielded from statistical attacks by the scheme's strong statistical features. The suggested technique, however, has a high sensitivity to both key and plaintext. The known-plaintext and chosen-plaintext attacks can be thwarted by this approach because the key stream and cipher frames are connected to the key and plain frames. The technique is trustworthy to be utilized for the secure video communication application, according to optimistic results.

Author contributions
All authors read approved the final manuscript.

Funding
Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB). No funding was received.

Availability of data and material
The datasets generated during and/or analyzing during the current study are available from the corresponding author on a reasonable request.

Declarations

Competing interests
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Received: 8 October 2022 Accepted: 21 March 2023
Published online: 06 April 2023

References

1. M. A. El-Mowafy, S. M. Gharghory, M. A. Abo-Elsoud, M. Obayya, and M. I. Fath Allah, *Chaos* Based encryption technique for compressed H264/AVC videos, IEEE Access **10**, 124002 (2022).
2. S. Chen, S. Yu, J. Lü, G. Chen, J. He, Design and FPGA-based realization of a chaotic secure video communication system. IEEE Trans. Circuits Syst. Video Technol. **28**, 2359 (2018)
3. M. Preishuber, T. Hütter, S. Katzenbeisser, A. Uhl, Depreciating motivation and empirical security analysis of chaos-based image and video encryption. IEEE Trans. Inf. Forensics Secur. **13**, 2137 (2018)
4. F. Liu, H. Koenig, A survey of video encryption algorithms. Comput. Secur. **29**, 3 (2010)
5. J. Yun, M. Kim, *JLVEA:* Lightweight real-time video stream encryption algorithm for internet of things, Sensors **20**, (2020).
6. H. M. M. Hosseini, P. M. Tan, Encryption of MPEG video streams, in *TENCON 2006 - 2006 IEEE Region 10 Conference* (2006), pp. 1–4.
7. F. Liu, H. Koenig, *Puzzle – A Novel Video Encryption Algorithm, in Communications and Multimedia Security* (Springer, Berlin, 2005), pp.88–97
8. Q. Wang, X. Wang, A new selective video encryption algorithm for the H.264 standard, in *2014 IEEE International Conference on Progress in Informatics and Computing* (2014), pp. 275–279.
9. H. Shen, L. Zhuo, Y. Zhao, An efficient motion reference structure based selective encryption algorithm for H.264 videos. IET Inf. Secur. **8**, 199 (2014)
10. Z. Wei, Y. Wu, R.H. Deng, X. Ding, A hybrid scheme for authenticating scalable video codestreams. IEEE Trans. Inf. Forensics Secur. **9**, 543 (2014)
11. T. E. Seidel, D. Socek, M. Sramka, *Cryptanalysis of video encryption algorithms*, in *Proceedings of the 3rd Central European Conference on Cryptology TATRACRYPT* (Bratislava, Slovak Republic, 2003).
12. B. Bhargava, C. Shi, S.-Y. Wang, MPEG video encryption algorithms. Multimed Tools Appl **24**, 57 (2004)
13. Y. Mei, Y. Jiang, Secure RFID system based on RC4 chaotic algorithm. J. Comput. Inf. Syst. **9**, 2083 (2013)
14. İ Öztürk, R. Kılıç, A novel method for producing pseudo random numbers from differential equation-based chaotic systems. Nonlinear Dyn. **80**, 1147 (2015)
15. J. Arif, M.A. Khan, B. Ghaleb, J. Ahmad, A. Munir, U. Rashid, A.Y. Al-Dubai, A novel chaotic permutation-substitution image encryption scheme based on logistic map and random substitution. IEEE Access **10**, 12966 (2022)
16. N.F.H. Ataa, R. Alawi, A proposal video encryption using light stream algorithm. Eng. Technol. J. **39**, 184 (2021)
17. S. Cheng, L. Wang, N. Ao, Q. Han, A selective video encryption scheme based on coding characteristics. *Symmetry (Basel)* **12**, (2020).
18. O. Benrhouma, A. B. Alkhodre, A. AlZahrani, A. Namoun, W. A. Bhat, Using singular value decomposition and chaotic maps for selective encryption of video feeds in smart traffic management. *Appl. Sci.* **12**, (2022).
19. W. M. Salama, M. H. Aly, Chaotic maps based video encryption: A new approach, in *2021 31st International Conference on Computer Theory and Applications (ICCTA)* (2021), pp. 18–25.
20. K. Kordov, G. Dimitrov, A new symmetric digital video encryption model. Cybernet. Inf. Technol. **21**, 50 (2021)
21. H. Xu, X. Tong, Z. Wang, M. Zhang, Y. Liu, J. Ma, Robust video encryption for H.264 compressed bitstream based on cross-coupled chaotic cipher. Multimed. Syst. **26**, 363 (2020)
22. M. Dua, D. Makhija, P. Y. L. Manasa, P. Mishra, 3D chaotic map-cosine transformation based approach to video encryption and decryption, **12**, 37 (2022).
23. A.T. Maolood, E.K. Gbashi, E.S. Mahmood, Novel lightweight video encryption method based on ChaCha20 stream cipher and hybrid chaotic map. Int. J. Electr. Comput. Eng. IJECE **12**, 4988 (2022)
24. A. Hafsa, M. Fradi, A. Sghaier, J. Malek, M. Machhout, Real-time video security system using chaos- improved advanced encryption standard (IAES). Multimed. Tools Appl. **81**, 2275 (2022)
25. U. Zia, M. McCartney, B. Scotney, J. Martinez, A. Sajjad, A novel pseudo-random number generator for IoT based on a coupled map lattice system using the generalised symmetric map. SN Appl. Sci. **4**, 48 (2022)
26. T. Sang, R. Wang, Y. Yan, Generating binary bernoulli sequences based on a class of even-symmetric chaotic maps. IEEE Trans. Commun. **49**, 620 (2001)
27. S. Bahrami, M. Naderi, Encryption of multimedia content in partial encryption scheme of DCT transform coefficients using a lightweight stream algorithm. Optik Int. J. Light Electron Opt. **124**, 3693 (2013)
28. E. Vaferi, R. Sabbaghi-Nadooshan, A new encryption algorithm for color images based on total chaotic shuffling scheme. Optik Stuttg **126**, 2474 (2015)
29. C. Chen, X. Wang, G. Liu, G. Huang, A robust selective encryption scheme for H265/HEVC video. IEEE Access 1 (2022).
30. N. Hemrajani, Novel selective video encryption for H.264 video. *Int. J. Inf. Secur. Sci.* **3**, 216 (2014).
31. Y. Wu, J. P. Noonan, and S. S. Agaian, NPCR and UACI randomness tests for image encryption. *Cyber J. Multidiscipl. J. Sci. Technol. J. Sel. Areas Telecommun. (JSAT) 31* (2011).
32. C. Boncelet, Chapter 7 - image noise models, in *The essential guide to image processing*. ed. by A. Bovik (Academic Press, Boston, 2009), pp.143–167

33. P. Patidar, M. Gupta, S. Srivastava, A.K. Nagawat, Image de-noising by various filters for different noise. Int. J. Comput. Appl. **9**, 45 (2010)
34. R. Garg, A. Kumar, Comparision of various noise removals using bayesian framework. Int. J. Mod. Eng. Res. IJMER **2**, 265 (2012)
35. H.S. Ranganath, S.G. Shiva, Correlation of adjacent pixels for multiple image registration. IEEE Trans. Comput. **C–34**, 674 (1985)
36. S. Fong, On improving the lightweight video encryption algorithms for real-time video transmission, in *2008 Third International Conference on Communications and Networking in China* (2008), pp. 1287–1293.
37. H. Elkamchouchi, W.M. Salama, Y. Abouelseoud, new video encryption schemes based on chaotic maps. IET Image Process **14**, 397 (2020)
38. T.-H. Le, J. Clediere, C. Serviere, J.-L. Lacoume, Noise reduction in side channel attack using fourth-order cumulant. IEEE Trans. Inf. Forensics Secur. **2**, 710 (2007)

Publisher's Note