Novel lightweight video encryption method based on ChaCha20 stream cipher and hybrid chaotic map

Abeer Tariq Maolood, Ekhlas Khalaf Gbashi, Eman Shakir Mahmood
Department of Computer Science, University of Technology, Baghdad, Iraq

Article Info

Article history:

Received May 26, 2021
Revised Jun 24, 2022
Accepted Jul 4, 2022

Keywords:

ChaCha20
Chaotic maps
Stream cipher
Video encryption
Video privacy

ABSTRACT

In the recent years, an increasing demand for securing visual resource-constrained devices become a challenging problem due to the characteristics of these devices. Visual resource-constrained devices are suffered from limited storage space and lower power for computation such as wireless sensors, internet protocol (IP) camera and smart cards. Consequently, to support and preserve the video privacy in video surveillance system, lightweight security methods are required instead of the existing traditional encryption methods. In this paper, a new light weight stream cipher method is presented and investigated for video encryption based on hybrid chaotic map and ChaCha20 algorithm. Two chaotic maps are employed for keys generation process in order to achieve permutation and encryption tasks, respectively. The frames sequences are encrypted-decrypted based on symmetric scheme with assist of ChaCha20 algorithm. The proposed lightweight stream cipher method has been tested on several video samples to confirm suitability and validation in term of encryption–decryption procedures. The performance evaluation metrics include visual test, histogram analysis, information entropy, correlation analysis and differential analysis. From the experimental results, the proposed lightweight encryption method exhibited a higher security with lower computation time compared with state-of-the-art encryption methods.

Corresponding Author:

Abeer Tariq Maolood
Department of Computer Science, University of Technology
52 Road, Al Senaa'h Street, Baghdad, Iraq
Email: abeer.t.maolood@uotechnology.edu.iq

1. INTRODUCTION

Recently, the encryption of multimedia data at the for front of research today, due to the continuous increment in digital communication on the internet and increasing uses of video in a wide range of applications security and privacy issues into serious attention. The nominal aim of multimedia data encryption is to make the multimedia information robust against the unauthorized divulgence in transit and storage. Many presented encryption schemes which are to preserve text data and image data are not adequate for video encryption because of the real time restriction and huge data volume in the video. Therefore, a lot of works have been presented for video encryption to improve the encryption quality and video security conditions based on utilizing different encryption algorithms. Hence, by comparing the chaos-based encryption with several encryption schemes, it has demonstrated an excellent performance with confirmed ability of increased security and privacy based on utilizing variable keys [1]–[8]. These make the chaotic scheme is adequate to implement video encryption for different applications. On the other hand, the ChaCha20 is stream cipher algorithm depended on XORed the plaintext with stream of pseudo-random bytes as a key of ChaCha20

[9]–[12]. ChaCha20 provides high confidence and much sensitive to change of the initial conditions, where a simple flipping of single bit into the input stream will because unpredictable changes at the output stream.

The problems of the previous schemes for securing the video data can be defined as, where these schemes were not taken into account the various parameters at simultaneously, such as quality of security, efficient computational with execution time, and compression efficiency, where many video-encryption schemes have been proposed to secure the local videos. Therefore, several of these schemes not adequate for secure the video transmission at real-world applications. From this point of issue, the motivation of this paper is to propose solution which can meet main requirements for secure transmitting the public and private videos based on a new encryption scheme by combined of chaos-based encryption models and ChaCha20 symmetric stream cipher algorithm. In this paper, the proposed scheme reduces the size of video data by extraction the differences of sequences frames based on dynamic reference frame. The resultant video will be double encrypted using two encryption phases, chaos-map and ChaCha20. The proposed scheme provides the main requirements for transmitting videos which are compression by reduces the video data, computational, quality of security by utilized double security phases, and efficient computations by utilizing two-model chaotic maps, first model to generate encryption keys to encrypt individual frame and second model to generate initial key of ChaCha20 algorithm. This paper is organized as follows: section 1 presents an introduction of this paper, section 2 presents the related work, section 3 presents the objectives and contributions of proposed video encryption scheme, the material and methods presented in section 4, the proposed video encryption scheme presented in section 5, the experimental results presented in section 6, the conclusion presented in section 7 at a final of this paper.

2. RELATED WORK

In this section, different related works on video encryptions are discussed and explained. The security performance analysis of video encryption schemes is presented and investigated in different research. In general, the video encryption was divided for selective encryption and full encryption, based on the amount of encrypted data and the requirements of the security level [13]–[18].

A selective video encryption scheme was presented to encrypt the interested part of the frame which of greatest information for the user, based on entropy measure of the frame data blocks and utilizing the chaotic map [19]. The presented, method depends on encrypted video by selecting interesting macro-blocks to be encrypted based on entropy measure. This method exhibited a strength security against an entropy-based attack, and it is suitable for video on demand applications. The encrypted video outflow this scheme has the same frames with distortion of several macro-blocks in each frame. In [20], a selective video encryption scheme based on coding characteristics was presented based on two case of encryption video. First case is complete selective video encryption using video encoding and 4D hyper-chaotic systems at the small amount of encrypted data. Second case is selective video encryption was implemented based on video characteristic encoding such as the motion vector difference (MVD), residual coefficient, intra prediction mode (IPM) and delta-QP in each slice of H.264/AVC encoding algorithm. This method presented high quality of encrypted video with high processing complexity for encrypted video. The encryption scheme was depended on three phases of work progress such as first phase is constructing plaintext, second phase is encrypting plaintext, and third phase is replacing the original bitstream. The H.264/AVC encoding algorithm was adopted to encode the video into independent multiple slices. The advanced encryption standard (AES) based cipher feedback mode (CFB) was utilized to encrypt these slices individually. The pseudo-random number generator (PRNG) was used to generate the required keys and real time updated.

On the other hand, a lot of researchers have been presented different schemes to encrypt whole video frame by adopting different encryption algorithms [21]–[26]. In work [24], to generate the key stream for video frames encryption, two algorithms were utilized different chaotic. The video file had been compressed before applied encryption process. The two proposed schemes were sensitive to slight variations in any components of comprising the secret encryption key as apparent from the obtained differential measures. In work [27], AES algorithm was adopted to encrypt video frame directly on the compressed domain with motion picture expert group (MPEG) video compression. In this method, the time cycle of compression and decompression was preserved.

In [28], the 2 dimension (2D) chaotic model was presented to apply encryption sachem for the multimedia encryption at high security data transmission. Their work was presented a hybrid chaotic structure based on multiple combined maps for different media encryption (text, voice, image, and video). At video encryption scheme of this method, the whole video frames were encrypted as separated image. This method exhibits high security data transmission with increases computation complexity and consuming in processing time.

In [29], a three-level chaotic video encryption scheme was presented based on employed the permutation and diffusion rounds. The maps of combination of logistic and tent (LTS) were utilized to obtain the prefatory encryption parameters. Their method exhibited good competency for processing time; however, the drawback was a compression reduction. The key generator block (KGB) was utilized to generate the internal keys of the key generation technique in order to generate required keys for chaos system. The internal keys were adopted to present the frame selection (FS) technique. On the bases of the presented results, the key space of the presented scheme was 2^{212} which can prove good security level. Also, the maximum deviation and deviation irregular were achieved good results compared with different related works.

In [30], the S-Box and two alternative schemes were utilized to present video encryption pipeline. These schemes are higher dimensional chaotic map and Ikeda delay differential equation (DDE). The associated limitations of this method were increasing the key complexity and processing time for encryption. In [31], the region of interest (ROI) based on faster region based convolutional neural networks (R-CNN) method for video encryption was presented to overcome the shortcoming of several video encryption schemes. Different encryption schemes had been used to encrypt the non-ROI in order to implement fully video frame encryption. This method presented an improved the performance of video encryption scheme based on reduces the amount of data in video for encryption at ROI. However, the main limitation of this method was increasing the encryption time and computation complexity by using two cases of encryptions (ROI and non-ROI) with different encryption scheme.

On the other hand, the encryption of video traffic had been implemented based on design of algorithms for identification/prediction attack from encrypted traffic. In [32], the practical mobile traffic had been designed based on utilized the deep learning as a viable strategy. This classifier was proposed to extract the features automatically and it has capability to deal with encrypted traffic, then reflecting their complex patter of traffic, while in [33]–[36], multi classifier have been proposed for mobile application traffic based on different approaches for instance Markov modeling, capture, and ground-truth creation. Finally, the general comparisons of the previous video encryption schemes that demonstrated in the literature are illustrated in Table 1.

Table 1. Comparisons of the previous video encryption schemes

Ref	Algorithm	Methods	Time	Encryption Ratio	Size
[31]	Hyper chaos and GF(17)	ROI and Fasted-CNN	0.2307 sec	100%	No change
[30]	Chaos system	Chaos maps and Ikeda time delay system	0.8062 sec	100%	No change
[29]	1D Chaotic Maps	Chaos system	0.5403 sec	100%	No change
[19]	chaotic map and entropy measure	Chaotic system	------	100%	
[16]	4D and 3D Arnold’s cat map and Chebyshev map	Arnold cat map	--------	100/%	No change

3. OBJECTIVES AND CONTRIBUTIONS OF PROPOSED VIDEO ENCRYPTION SCHEME

Many researchers’ literature in this paper had been interested on the video encryption issue in order to validate video encryption in real time by utilizing different techniques. Their works had been associated with different weakness points. It is noteworthy, several video-encryption algorithms had been developed to secure the local videos, but these encrypted videos are not appropriate for transmitting in real-world applications. The main contribution of this paper, a new video encryption scheme has been proposed to shorthand the computation time by reduction the data included in the frame. This contribution is carried out based on computing the differences between sequences frames, in order to produce video frames, have less video data than original video with availability to return original video at decryption end. The presented work in this paper aims to improve the encryption quality, reduces processing time, reduces the execution time, and increases the robustness against different attacks. In order to achieve these objectives, the proposed work in this paper is adopted combinations of different algorithms to present light video encryption scheme which are Henon chaotic map, ChaCha20 algorithm, and algorithm of differences of sequences frame. Finally, to validate the mentioned aims of this paper, the proposed video encryption scheme will be tested by different video files. Then, the finding of the experimental results will be described in sufficient details.

4. METHODS AND MATERIALS

The chaotic map and ChaCha20 algorithms are utilized in this paper to propose a new video encryption scheme for real time applications. In order to implement the proposed video encryption scheme in this paper, efficient secret keys are needed to obtain a robust and powerful encryption model against variant attacks. Accordingly, hybrid chaotic maps are adopted and carried out in two phases models to generate the secrete keys in this work. First chaotic model is employed to generate chaotic sequence-based permutation process of light-weight encryption scheme. The second chaotic model is exploited to generate the initial keys of Chach20 method as will be explained in algorithm 1.

Algorithm 1: keys generation -based encryption

```
Input: a, b, Xo, Yo, Mid, M
Output: Key2[M] // list of generated keys
     Define basic model parameters: a and b
     Initialize model parameters (Xo and Yo)
     Start Iteration Mid
     Update XMid+1 and YMid+1
     End iteration Mid
     Start iteration M
     X,Y ← update XM+1 and YM+1 ///Generate encryption keys using (2.a) and (2.b)
     Key[M]=select1(X, Y, 32)
     Key1[M]=select2(Key(M),16)
     Key2[M]=integrate(Key, Key1)
End iteration M
```

4.1. Hybrid chaos models-based keys generation

Henon chaotic map presented in [37] is adopted in the first model (model 1) to generate two-dimension chaotic map Xi, Yi where (i=1, 2, ..., M) and used to perform the permutation process in the lightweight encryption phase.

$$X_{i+1}=1+b_1Y_i-a_1X_i^2 \tag{1a}$$

$$Y_{i+1}=X_i \tag{1b}$$

Where a_1, b_1 are the initial parameters of this model with initial values a1 ∈ [1.07, 1.4] and b_1=0.3. Xo_1, Yo_1 initial values are stand to Xo_1=0.5, Yo_1=0.5. It is worth noting that the first model of Henon chaotic maps is iterated by $(H\times W)+Mid$, where H, W refereed to high and width of the current frame, respectively. Mid is a predefined iteration number. The second model is iterated by M times. The second model (model 2) of Henon chaotic map presented in [6] is adopted to generate the important initial key of chach20 encryption method. The initial values of this model are, a_2=1.4, b_2=0.3, Xo_2=1.2, Yo_2=0.8 as illustrated in (2):

$$X_{i+1}=1-a_2X_i^2+Y_i \tag{2a}$$

$$Y_{i+1}=b_2X_i \tag{2b}$$

where both functions select1 () and select2 () are implemented for selecting 32, 16 bytes from the original chaos keys X, Y, respectively. Then we integrate the obtained keys of each iteration to construct the final key Key2 used later for chach20 encryption method.

4.2. ChaCha20 encryption algorithm

ChaCha20 algorithm is stream cipher has naturally good performance, as well as itis fast, secure, and simple encryption algorithm. ChaCha20 implemented in this paper is derived from the code and algorithms available in different research works and reports [9]–[12]. Its input includes a 48 byte consists of 32-byte key, a 4-byte counter, a 12-byte nonce. The core is a pseudo-random number generator. The output cipher data is obtained by XOR'ing the input plain data with a pseudo-random stream: $Cipher - data = input\ plain - data\ XOR\ ChaCha20 - stream\ (key, nonce)$, at decryption, the same operation as encryption: $plain\_data = cipher\_data\ XOR\ ChaCha20\_stream\ (key, nonce)$.

The ChaCha20 ciphering process is depended on 20 rounds of mathematical calculations using XOR, addition and rotation using as inputs four 32-byte key, a 4-byte counter, a 12-byte nonce as input matrix as shown in Table 2. The number of rounds depends on the application requirements such as 20 round (maximum security), 8 rounds (maximum speed) or 12 rounds (balance between speed and security) [9]. In our work, the 20 round of ChaCha20 algorithm is implemented to provide high data security for the transmitted video.

Table 2. Input matrix

Const.	Const	Const	Const
Key	Key	Key	Key
Key	Key	Key	Key
Counter	Nonce	Nonce	Nonce

5. THE PROPOSED VIDEO ENCRYPTION SCHEME

This section describes the proposed encryption/decryption schemes for the acquired plain video sequence Pv={F1, F2, ……Fn}. The main steps of the proposed encryption algorithm are schematized in Figure 1 and illustrated in algorithm 2. Likewise, the main steps of the proposed decryption algorithm are demonstrated in Figure 2 and stated in algorithm 3. The decryption phase is designed and implemented to reconstruct the original video from the encrypted sequenced frames.

Algorithm 2: the proposed video encryption scheme

```
Input: Plain video Pv, Key2, $a_1$, $b_1$[1.07, 1.4], $Xo_1$= 0.5, $Yo_1$=0.5, $a_2$ = 1.4, $b_2$ = 0.3,
       $Xo_2$= 1.2, $Yo_2$ =0.8, step
Output: Cipher video $C_v$
Begin
    $X_1$,$Y_1$ ← Key Generation-based Permutation (a1, b1, Xo1, Yo1) // using model
    Ke2, $X_2$,$Y_2$ ← Key Generation-based Encryption ($a_2$, $b_2$, $Xo_2$, $Yo_2$) // using model 2
    Len=size(Pv)
    Ref=$P_v$(1)  // taken reference frame from $P_v$
    Set i=1
    while i+1<Len-step
    Read (Pv, Fi)  // read frames from Pv
    DFi=Pre-processing (Fi, Ref) // find dissimilar frames using Algorithm (2)
    LDFi=Lightweight Encryption (DFi, X1, Y1) // using Algorithm (4)
    EPvi=Chacha-Encryption(LDFi, Key2)
    Cv[i]=EPvi     //Construct array of encrypted frames
    if i% step==0  Ref=Fi  end if
 End while
End
```

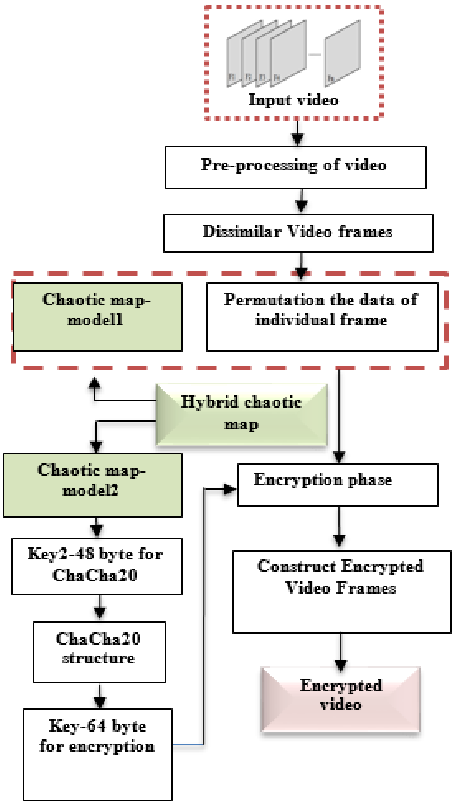


Figure 1. Diagram of the proposed video encryption algorithm

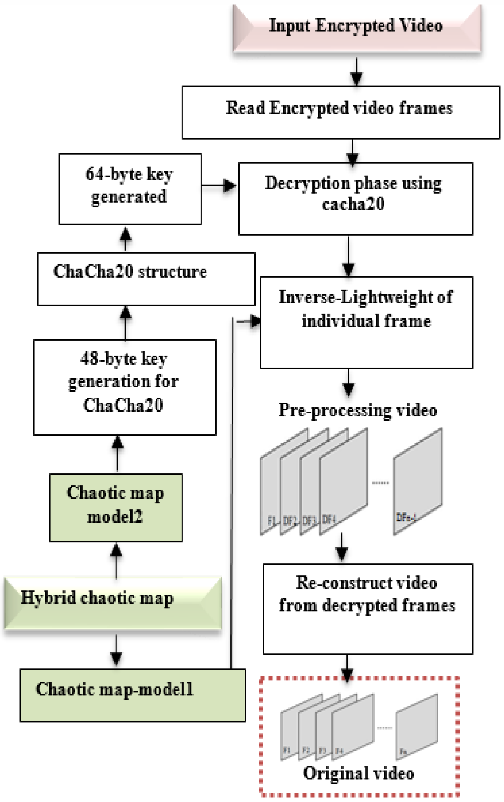


Figure 2. Diagram of the proposed video decryption algorithm

Algorithm 3: video decryption scheme

```
 Input: Cipher video Cv, Key2, X, Y, step
 Output: Palin video Pv
Begin
```

```
   Len=size(Cv)
   Ref=Cv(1)
   While i+1<Len
     Read (Cv, Fi)  // read one frames from $P_v$
     DCvi=Chacha-Decryption (Fi, Key2)  // decryption phase
     LDFi=Inverse-Lightweight (DCvi, X, Y) // inverse-lightweight phase according to model1
     DFi=Pre-processing (LDFi, Ref)
     $P_v$[i]=DFi   //Construct encrypted video from encrypted frames
     if i>=step then
         Ref=Fi
            end if
          End while
End
```

5.1. Pre-processing of video using frames differences

In this section, the frame differences technique will be explained. This technique is applied at both encryption and decryption sides in order to reduce the processing time. The main objective of utilizing frames differences technique is to estimate the changes between sequenced frames (reference frame and current frame) along video sample based on their dissimilarity measurement. In this context, the reference frame is updated at a predefined interval of frames named step. Afterwards, the output video obtained from pre-processing phase is passed into lightweight encryption phase with lower spatial information of each video frame in order to minimize the computation time of the encryption/decryption processes. The main steps of pre-processing approach are illustrated in Figure 3. The main contributions of adopting frame differences as pre-processing phase in our video encryption scheme can be summarized as follows: i) minimize the spatial information of video frame, ii) reduces the processing time and complexity of preprocessed video in encryption and decryption sides, and iii) lower redundancy of video frames are acquired through adopting step size for reference frame denoting.

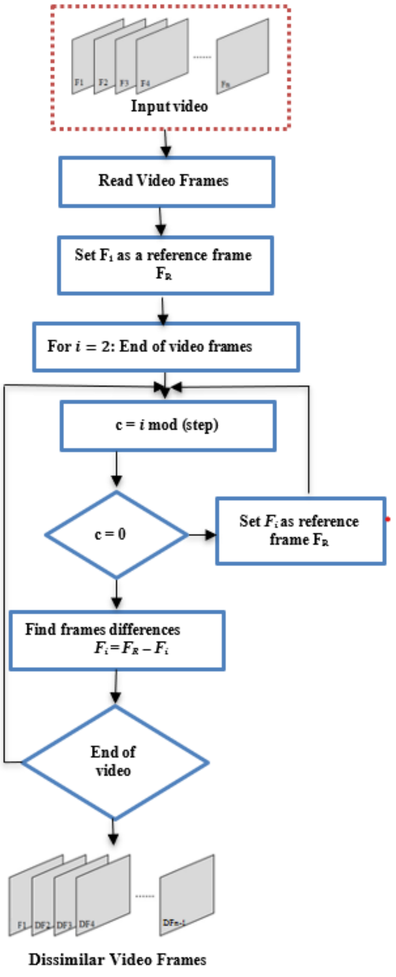

F1
F2
F3
F4
Fn
Input video
Read Video Frames
Set F1 as a reference frame FR
For i = 2: End of video frames
c = i mod (step)
c = 0
Set Fi as reference frame FR
Find frames differences Fi=FR – Fi
End of video
F1
DF2
DF3
DF4
DFn-1
Dissimilar Video Frames

Figure 3. Diagram of pre-processing approach

5.2. Lightweight encryption scheme

A lightweight encryption scheme is designed and implemented to encrypt dissimilar video frames obtained from pre-processing video phase. The core of this algorithm is carrying out the data permutation for individual frame based on the chaotic map model (1). The aim of lightweight encryption algorithm is to make the encryption complexity is strong resistance against most measurable tests. The lightweight encryption algorithm is illustrated in algorithm 4.

Algorithm 4: lightweight encryption

```
 Input: Dissimilar video frame DFi, X1, Y1
 Output: Cipher video frame LDFi
   Begin
      Examine the frame dimensions (DFi) and reshape into one dimensional vector of pixels
      (DFvi=H×W×3)  // H, W are height and width of DFi video frame)
      Permute DFvi data frame (pixels) using chaotic sequence $X_1$, $Y_1$  // permutation phase
      according to model1
      $LDF_i$=reshape (DFvi, H, W)
   End
```

6. RESULTS AND DESCUSSION

The experimental results of many video files are tested to validate and investigate the objectives are mentioned in this work. For this purpose, the proposed encryption scheme and the key generation algorithm have been investigated on the platform (MATLAB 2018, windows 8.1-64-bit, CPU Cor i7, RAM 6 GB, 2.2 GHz) and tested with different video files. The strength and suitability of proposed scheme for video encryption have been tested for key sensitivity, key space, computational time, randomness, and resistance against statistical attacks.

The experimental results are demonstrated for different video files listed from any video dataset. For the simulation experiments, the selected video files have properties are enlisted in Table 3. The important key metrics are measured to validate the objectives of the proposed video encryption scheme. The following progresses are implemented to establish the validation of the proposed scheme.

Table 3. Properties of video files used in simulation results

Attribute	Value
Number of frames	821
Height	640 pixels
Width	360 pixels
Frame rate	25-FPS
Bit per pixel	24-BPP
Video format	RGB24

6.1. Visual degradation measurements

To measure the perceptual distortion of the encrypted video frame compared with original video frame, the visual degradation must be measured. The simulation results for selected video frames of different videos are illustrated in Figure 4 (Figures 4(a) and 4(b)), where these videos have properties listed in Table 3. From this figure, it can be inferred that, the proposed encryption scheme offers high scale of visual degradation. Therefore, the proposed encryption/decryption scheme as shown in Figures 4(c) and 4(d) respectively is very suitable to encrypt high class data-based multimedia applications.

6.2. Key space analysis

In crypto analysis system, the Brute force attack depends on trying out all possible keys to recover the original video data. The size of key space for the encryption scheme provides its strength to make Brute-force attack unfeasible. In this work two dimensional Henon chaotic systems were employed to provide a larger key space. The 2D chaotic systems are utilized to generate the secret key for two encryption phases of the proposed video encryption scheme. The encryption values of ChaCha20 depend on its initial seed keys (key, nonce, counter and constants) and its related parameters associated with the second model of Henon map. Likewise, the initial encryption values of light weight scheme depend upon the first model of Henon map equation and its initial values. In addition, the key space depends upon the number of decimal places of the mantissa which is approximately 2^{15}×2^{15}=230 for each of model1 and model2 which yields 2^{60} key spaces based chaotic map generation. This making the Brute force attack is computationally infeasible. This is a main contribution of using two Henon chaotic maps to realize a larger key space. Further, the key space regards the encryption phase based on ChaCha20 is denoted as 2^{348}.

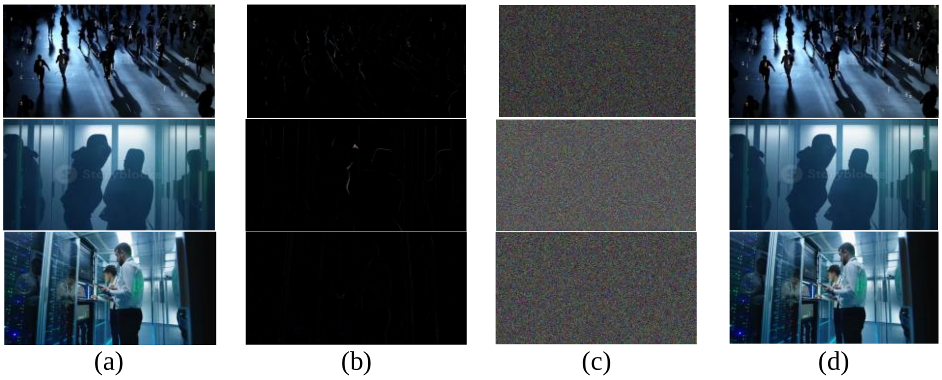


Figure 4. Simulation results of proposed encryption scheme for selected frame taken from different videos: (a) original frame, (b) differences frame, (c) encrypted frame, and (d) decrypted frame

6.3. Correlation coefficients analysis

The correlation coefficient reflects the pixels distribution and their relationships (dependency or independency) in the adjacent video frames. This mean, the relationship between adjacent frames indicates correlation value. The correlation is high when the correlation value is (approach to 1) at the linear relationship, while correlation is implying low when the correlation value (approach to zero) at the nonlinear relationship [8]. Therefore, the correlation value between adjacent frames of encrypted video must be minimized. The correlation between adjacent frames is measured for the pixels located at the same position of adjacent frames. In this paper, the differences between adjacent frames are adopted to provide a new video frame with minimum data size. This technique makes the correlation value (approach to zero) and the relationship between adjacent frames is nonlinear.

The experimental results of the correlation coefficients related to adjacent frames for the original and encrypted frames are illustrated in Table 4. From these results, it is inferred that there exists negligible correlation between original and encrypted frames, thereby providing no clue to make the statistical cryptanalysis is possible. A scientific comparison was conducted between the proposed encryption scheme and published research work presents in literature to evaluate the performance of the proposed encryption scheme. This comparison has been implemented based on the properties of video file “dfs.avi”, which is utilized in [14] and illustrated in Table 5. The comparison simulation results are illustrated in Table 6.

Table 4. Simulation results of correlation coefficient

Correlated Frame No.		Correlation coefficient of red component			Correlation coefficient of green component			Correlation coefficient of blue component		
		Original	Differences	Encrypted	Original	Differences	Encrypted	Original	Differences	Encrypted
1	1	1	1	1	1	1	1	1	1	1
1	2	0.9951	0.1077	-0.00027	0.9627	0.1065	-0.00053	0.9959	0.1106	-0.00074
2	3	0.9950	0.4121	-0.00049	0.9643	0.3800	-0.00310	0.9959	0.4276	-0.00043
3	4	0.9939	0.0219	0.00013	0.9670	0.0066	0.00051	0.9949	0.0089	0.00064
4	5	0.9932	0.0788	0.00025	0.9645	0.0570	0.00033	0.9942	0.0647	0.00073
5	6	0.9328	0.2263	0.00011	0.9610	0.2450	0.00201	0.9938	0.2348	0.00047

Table 5. Properties of video files “dfs.avi” [14]

Attribute	Value
Number of frames	102
Height	240 pixels
Width	320 pixels
Frame rate	15-FPS
Bit per pixel	24-BPP
Video format	RGB24

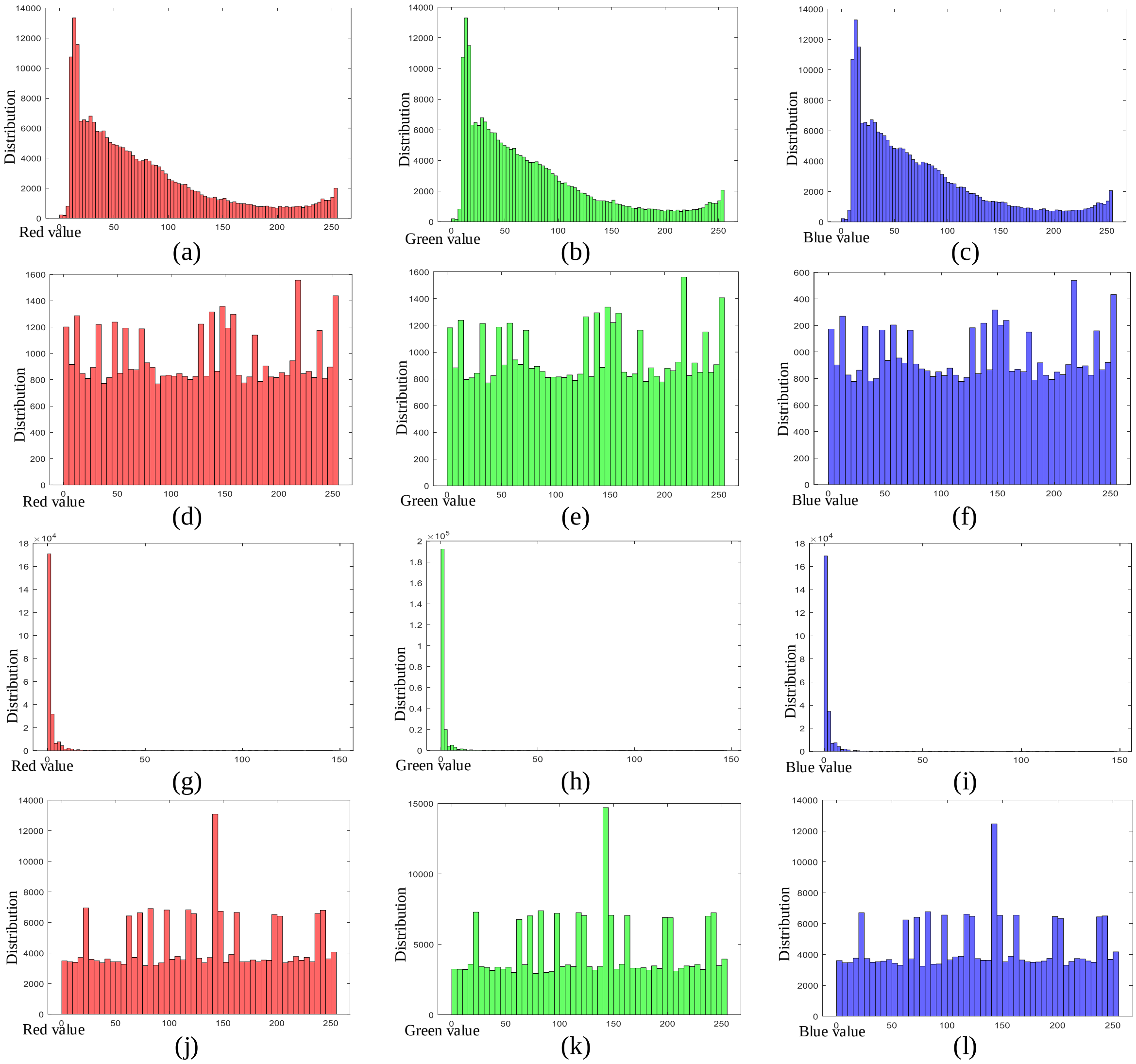
6.4. Histogram analysis

The histogram analysis reveals the graphical relationships between the original and encrypted frame based on the frame pixels distribution. Statistical attacks are made by exploiting the predictable this relationship to recover the original frames. The considerable changes in the histograms of encrypted frames compared with the original frames reflect the strength of the encryption scheme against statistical attacks.

The histograms of the original frames are shown in Figure 5 (Figures 5(a) to 5(c)) and encrypted frames are illustrated in Figures 5(d) to 5(f), for the red, green, and blue components. Obviously, the histograms of the encrypted frame for three components red, green, blue are uniformly distributed and significantly different from the original frames. Subsequently, they do not provide any evidence for exploiting the statistical attack. Also, the differences frames are shown in Figures 5(g) to 5(i) and encrypted differences frames are illustrated in Figures 5(j) to 5(l), for the red, green, and blue components.

Table 6. Comparison simulation results of correlation coefficient

Frame No.	Frame No.	Correlation coefficient of red component		Correlation coefficient of green component		Correlation coefficient of blue component	
		Ref [14]	Proposed method	Ref [14]	Proposed method	Ref [14]	Proposed method
1	2	− 0.0065	-0.0082	− 0.0066	−0.0072	−0.0020	-0.0062
2	2	1	1	1	1	1	1
2	3	− 0.00740	-0.00930	− 0.0067	− 0.0085	0.0089	0.00036
3	4	0.00022	0.00012	0.0037	− 0.00042	0.0087	0.00042
4	5	0.0014	-0.00250	0.0036	0.00031	0.00087	0.00023



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Figure 5. Histogram components (red, green, blue) of original frame, differences frame and encrypted frame for (original and differences frame): (a) original frame (red), (b) original frame (green), (c) original frame (blue), (d) encrypted original frame (red), (e) encrypted original frame (green), (f) encrypted original frame (blue), (g) differences frame (red), (h) differences frame (green), (i) differences frame (blue), (j) encrypted differences frame (red), (k) encrypted differences frame (green), and (l) encrypted differences frame (blue)

6.5. Randomness test

The randomness analysis is required in video cryptographic by measuring the entropy of each frame at both original and encrypted video frames. From a security viewpoint, the entropy for the encrypted frame should be close to an ideal value. Significantly, the higher values of entropy indicate results in high randomness (statistically), and high distortion results in the encrypted frames.

$$Entropy = -\sum_{1}^{n}(pi\log(pi)) \tag{3}$$

Where pi ∈ [0.0, 1.0] and −log pi represents the information associated with a single occurrence of pi. Using (3) [19], entropy is found for selected frame taken from original, encrypted, and decrypted frames of the video. The simulation results and comparison are illustrated in Table 7. From the simulation results, the entropy values obtained are close to an ideal value of eight. Therefore, the proposed scheme provides an encrypted video has strength against an entropy-based attack due to lower effectiveness of this type of attackers when the data outflow is minimized.

Table 7. Simulation results and comparison of entropy (randomness)

	Proposed Scheme	Ref [15]
The entropy of the original frame	7.9996	7.4541
The entropy of the encrypted frame	7.9989	7.4535
The entropy of the decrypted frame	7.9987	7.4537

6.6. Differential attacks analysis

Differential attack could be called the choice of plaintext attack, where it is utilized to measure the influence of modification one pixel with respect to whole video frame. For this purpose, two measures (NPCR and UACI) can be adopted to evaluate and measure the differential attack. These two measures can be computed using mathematical formulas explained in (4) and (5) [19]:

$$NPCR = \sum_{i=1}^{H}\sum_{j=1}^{W} D(i,j)\frac{100}{H\times W} \tag{4}$$

$$UACI = \sum_{i=1}^{H}\sum_{j=1}^{W} \frac{|F1(i,j)-F2\,(i,j)|}{255}\frac{100}{H\times W} \tag{5}$$

where *F1(i, j)* and *F2(i, j)* are the pixel intensity of two encrypted video frames F1 and F2 related to the original video frames. The symbol (H) signifies the row count and (W) signifies the column count.

The *D(i, j)=0* if encrypted *F1(i, j)=original F1(i, j)*, otherwise it is equal to 1. Table 8 demonstrates that the results of differential attack analysis for randomly chosen video frames. These results show that the encrypted frame is 99.7% than the original video frames. Table 8 demonstrate that the results of differential attack analysis for randomly chosen video frames. These results show that the encrypted frame is 99.7% than the original video frames. Another type of comparison related to *NPCR* and *UACI* metrics was performed which demonstrates the percentage of *NPCR* and *UACI* for different video samples as shown in Table 9.

Table 8. Simulation results of differential attacks

Sample frame number	Video frame chosen	NPCR %	UACI %
1	1	99.3	33.71
2	10	99.7	33.68
3	15	99.56	33.69
4	20	99.7	33.73

Table 9. Comparison results of NPCR and UACI metric

Test	Ref [28]		Ref [29]		Proposed	
video	*NPCR*	*UACI*	*NPCR*	*UACI*	*NPCR*	*UACI*
Rhino	99.61	33.61	99.51	33.54	99.73	33.68
Flamingo	99.63	33.63	99.52	33.52	99.68	33.77
Train	99.63	33.63	99.61	33.50	99.71	33.78
VIP-train	99.61	33.60	99.58	33.62	99.75	33.67

6.7. Encryption speed time and encryption ratio analysis

In real time security applications, the encryption speed is a critical key to implement video encryption scheme. Therefore, to increase the encryption speed of the encryption scheme, encryption ratio should be minimized. Encryption ratio denoted as the ratio between the size of the encrypted frame and the original frame size. Achieving the equilibrium between the strength of encryption and encryption speed by completely encrypting the entire video frame and increases the encryption speed as well as adopting the differences of video frames technique which proposed in this work as illustrated in pre-processing phase of this work.

The proposed scheme significantly reduces the time of encryption and decryption the whole video by eliminating the similar information between the sequences frames making it suitable for real time video encryption. In Tables 10 and 11, we have stated a comparison result related to consuming time of the proposed scheme against literature research works. Obviously, the obtained results demonstrate that the proposed video encryption scheme has less time consuming and faster compared to video encryption schemes listed in in literature. It is owing to; our encryption scheme is based essentially on encryption the data differences between sequences frames in video instead of encryption the whole video frames.

Table 10. A comparison results of encryption speed time (in frame/seconds) for randomly selected frames

Test video	Ref [30] 8D	Ref [30] 12D	Ref [30] IKeda	Ref [29]	Proposed scheme
Rhino	0.9124	1.7964	1.2147	0.5403	0.2311
Flamingo	0.8062	0.8402	1.1124	0.5982	0.2503
Train	0.8631	1.6493	0.9908	0.5178	0.1926
VIP-train	1.0588	2.3137	1.3574	0.4723	0.16821

Table 11. A comparison results of encryption time (second) per frame

Test video		Ref [29] (Time (second))	Proposed (Time (second))
Grandma		0.2927	0.0341
Akiyo		0.2994	0.0344
Foreman		0.3847	0.0427

6.8. Computational complexity of the proposed lightweight encryption scheme

This section discusses the computational complexity of the proposed lightweight encryption scheme. Encryption and decryption of whole video frame is a one of the big problems to applied real time cryptographic scheme at different video types. The proposed scheme in this paper contribute to mitigate this problem by reduces the computational complexity of encryption scheme based on reduces the number of total data points for each video frame through adopting the frame differences technique. This technique led to make computational complexity of the cryptographic scheme is a light weight. Finally, the proposed encryption scheme present reduces the computational complexity of encryption and improves the encryption speed.

7. CONCLUSION

In this paper, we have proposed, presented, and investigated a new light weight stream cipher scheme of video frames based on hybrid chaotic system and ChaCha20 algorithm. A lightweight encryption scheme-based permutation process is employed to scramble video frames contents using chaotic system-model 1. We have used chaotic system for permutation task due to its robustness, randomness, and it is faster than traditional scheme for stream ciphers. Thus, the chaos system was utilized to generate encryption keys at the lightweight encryption phase and to generate the seed keys of ChaCha20 encryption algorithm. In addition, to provide a larger key space, 2D chaotic systems have been employed to generate the secret keys for two encryption phases of the proposed video encryption scheme. The permutation process associated with the proposed encryption scheme is contributed to scramble the frame pixels based on chaos key streams to provide the randomness distribution of the pixels' positions. Our experimental results of UACI and NPCR metrics have been shown that any small changes in the original video frame can be spread overall pixels in the cipher video frame. The ChaCha20 stream cipher algorithm is implemented with 20 rounds (maximum secure) in order to produces high quality, lightest and significant performance improvements of encryption scheme. The ChaCha20 combined with chaos maps are utilized to provide robust video encryption scheme to send public and private videos on unsecure communication network.

Further, we have presented a new technique for the purpose of encryption video contents through exploiting the difference criteria between each sequences frames in order to minimize the computation time, getting minimal correlation coefficient as well as obtaining image histogram with uniformly distribution. As we shown in the experiments, the proposed method is more robust against the statistical attacks. Further, the results of histograms and correlation coefficients proved that the proposed encryption scheme is resistance against statistical attacks. Finally, the advantages of the proposed scheme have been highlighted by comparing it against different state-of-the-arts methods from literature. Based on the acquired comparative performance results, the proposed scheme is obviously has extra efficacious in term of entropy, correlation coefficients, NPCR and UACI metrics and encryption time(s). The suggested future works are implementing the proposed encryption and decryption scheme using FPGA technique. Also, apply the proposed scheme in this paper on the high quality 4k format for both image and video frames. Finally, the region of interest extraction (ROI) will be utilized to implements the proposed scheme in order to reduce the important data points of video frames. This leads to reduce the computation time and the computational complexity of video encryption scheme, as well as making the encrypted video is very adequate for real time applications.

ACKNOWLEDGEMENTS

This article is supported by University of Technology/Computer Science Department, Baghdad, Iraq.

REFERENCES

[1] M. S. Azzaz, C. Tanougast, S. Sadoudi, and A. Bouridane, “Synchronized hybrid chaotic generators: Application to real-time wireless speech encryption,” *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 8, pp. 2035–2047, Aug. 2013, doi: 10.1016/j.cnsns.2012.12.018.

[2] A. Hasheminejad and M. J. Rostami, “A novel bit level multiphase algorithm for image encryption based on PWLCM chaotic map,” *Optik*, vol. 184, pp. 205–213, May 2019, doi: 10.1016/j.ijleo.2019.03.065.

[3] J. Yu, S. Guo, X. Song, Y. Xie, and E. Wang, “Image parallel encryption technology based on sequence generator and chaotic measurement matrix,” *Entropy*, vol. 22, no. 1, pp. 76–81, Jan. 2020, doi: 10.3390/e22010076.

[4] B. Yousif, F. Khalifa, A. Makram, and A. Takieldeen, “A novel image encryption/decryption scheme based on integrating multiple chaotic maps,” *AIP Advances*, vol. 10, no. 7, Jul. 2020, doi: 10.1063/5.0009225.

[5] M. A. Murillo-Escobar, M. O. Meranza-Castillón, R. M. López-Gutiérrez, and C. Cruz-Hernández, “A chaoticencryption algorithm for image privacy based on two pseudorandomly enhanced logistic maps,” in *Multimedia Security Using Chaotic Maps: Principles and Methodologies*, Springer:, Germany, 2020, pp. 111–136.

[6] M. Suneel, “Cryptographic pseudo-random sequences from the chaotic henon map,” *Indian Academy of Sciences*, vol. 34, no. 5, pp. 689–701, Apr. 2006.

[7] M. García-Martínez and E. Campos-Cantón, “Pseudo-random bit generator based on multi-modal maps,” *Nonlinear Dynamics*, vol. 82, no. 4, pp. 2119–2131, Dec. 2015, doi: 10.1007/s11071-015-2303-y.

[8] F. A. Elizalde-Canales, I. D. J. Rivas-Cambero, L. F. Rebolledo-Herrera, and C. J. Camacho-Bello, “Pseudo-random bit generator using chaotic seed for cryptographic algorithm in data protection of electric power consumption,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 2, pp. 1399–1409, Apr. 2019, doi: 10.11591/ijece.v9i2.pp1399-1409.

[9] M. Goll and S. Gueron, “Vectorization on ChaCha stream cipher,” in *2014 11th International Conference on Information Technology: New Generations*, Apr. 2014, pp. 612–615, doi: 10.1109/ITNG.2014.33.

[10] L. E. Kane, J. J. Chen, R. Thomas, V. Liu, and M. Mckague, “Security and performance in IoT: A balancing Act,” *IEEE Access*, vol. 8, pp. 121969–121986, 2020, doi: 10.1109/ACCESS.2020.3007536.

[11] P. McLaren, W. J. Buchanan, G. Russell, and Z. Tan, “Deriving ChaCha20 key streams from targeted memory analysis,” *Journal of Information Security and Applications*, vol. 48, Oct. 2019, doi: 10.1016/j.jisa.2019.102372.

[12] R. Velea, F. Gurzau, L. Margarit, I. Bica, and V.-V. Patriciu, “Performance of parallel ChaCha20 stream cipher,” in *2016 IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, May 2016, pp. 391–396, doi: 10.1109/SACI.2016.7507408.

[13] H. Schwarz, D. Marpe, and T. Wiegand, “Overview of the scalable video coding extension of the H.264/AVC standard,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007, doi: 10.1109/TCSVT.2007.905532.

[14] L. S. Lian, *Multimedia content encryption: techniques and applications*. New York, NY, USA: Auerbach Publications, 2008.

[15] F. Sbiaa, S. Kotel, M. Zeghid, R. Tourki, M. Machhout, and A. Baganne, “A selective encryption scheme with multiple security levels for the H.264/AVC video coding standard,” in *2016 IEEE International Conference on Computer and Information Technology (CIT)*, Dec. 2016, pp. 391–398, doi: 10.1109/CIT.2016.53.

[16] J. S. Bowade, P. Khade, Raghuwansh, and D. M. M., “Technique of video encryption/scrambling using chaotic functions and analysis,” *Journal of Emerging Technologies and Innovative Research (JETIR)*, vol. 2, no. 6, pp. 1951–1958.

[17] S. Lian, J. Sun, J. Wang, and Z. Wang, “A chaotic stream cipher and the usage in video protection,” *Chaos, Solitons & Fractals*, vol. 34, no. 3, pp. 851–859, Nov. 2007, doi: 10.1016/j.chaos.2006.03.120.

[18] B. Jovanović and S. Gajin, “An efficient mechanism of cryptographic synchronization within selectively encrypted H.265/HEVC video stream,” *Multimedia Tools and Applications*, vol. 77, no. 2, pp. 1537–1553, Jan. 2018, doi: 10.1007/s11042-017-4389-3.

[19] R. Malladar and R. Sanjeev Kunte, “Selective video encryption based onentropy measure,” in *Integrated Intelligent Computing, Communication and Security*, 2019, pp. 603–612.

[20] S. Cheng, L. Wang, N. Ao, and Q. Han, “A selective video encryption scheme based on coding characteristics,” *Symmetry*, vol. 12, no. 3, pp. 332–348, Feb. 2020, doi: 10.3390/sym12030332.

[21] Z. Lin, S. Yu, J. Lu, S. Cai, and G. Chen, “Design and ARM-embedded implementation of a chaotic map-based real-time secure video communication system,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 7, pp. 1203–1216, Jul. 2015, doi: 10.1109/TCSVT.2014.2369711.

[22] A. Alfalou, C. Brosseau, and N. Abdallah, “Simultaneous compression and encryption of color video images,” *Optics Communications*, vol. 338, pp. 371–379, Mar. 2015, doi: 10.1016/j.optcom.2014.10.020.

[23] J. Chen and F. Peng, “A perceptual encryption scheme for HEVC video with lossless compression,” *International Journal of Digital Crime and Forensics*, vol. 10, no. 1, pp. 67–78, Jan. 2018, doi: 10.4018/IJDCF.2018010106.

[24] H. Elkamchouchi, W. M. Salama, and Y. Abouelseoud, “New video encryption schemes based on chaotic maps,” *IET Image Processing*, vol. 14, no. 2, pp. 397–406, Feb. 2020, doi: 10.1049/iet-ipr.2018.5250.

[25] M. Eid, E.-S. M. El-kenawy, and A. Ibrahim, “A fast real-time video encryption/decryption technique based on hybrid chaotic maps,” *Journal of Computer Science and Information Systems*, vol. 2, no. 2, pp. 1–8.

[26] F. Yan, A. M. Iliyasu, S. E. Venegas-Andraca, and H. Yang, “Video encryption and decryption on quantum computers,”

International Journal of Theoretical Physics, vol. 54, no. 8, pp. 2893–2904, Aug. 2015, doi: 10.1007/s10773-015-2524-3.

[27] S. Keshav and A. B. Deshmukh, “Video frame encryption algorithm using AES,” *International Journal of Engineering Research & Technology (IJERT)*, vol. 5, no. 6, pp. 588–591, 2016.

[28] I. Yasser, M. A. Mohamed, A. S. Samra, and F. Khalifa, “A chaotic-based encryption/decryption framework for secure multimedia communications,” *Entropy*, vol. 22, no. 11, pp. 1253–1276, Nov. 2020, doi: 10.3390/e22111253.

[29] R. Ranjith kumar, D. Ganeshkumar, A. Suresh, and K. Manigandan, “A new one round video encryption scheme based on 1D chaotic maps,” in *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*, Mar. 2019, pp. 439–444, doi: 10.1109/ICACCS.2019.8728443.

[30] D. Valli and K. Ganesan, “Chaos based video encryption using maps and Ikeda time delay system,” *The European Physical Journal Plus*, vol. 132, no. 12, p. 542, Dec. 2017, doi: 10.1140/epjp/i2017-11819-7.

[31] L. Duan, D. Zhang, F. Xu, and G. Cui, “A novel video encryption method based on faster R-CNN,” in *Proceedings of the 2018 International Conference on Computer Science, Electronics and Communication Engineering (CSECE 2018)*, 2018, pp. 100–104, doi: 10.2991/csece-18.2018.21.

[32] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescape, “Mobile encrypted traffic classification using deep learning: Experimental evaluation, lessons learned, and challenges,” *IEEE Transactions on Network and Service Management*, vol. 16, no. 2, pp. 445–458, Jun. 2019, doi: 10.1109/TNSM.2019.2899085.

[33] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, “Multi-classification approaches for classifying mobile app traffic,” *Journal of Network and Computer Applications*, vol. 103, pp. 131–145, Feb. 2018, doi: 10.1016/j.jnca.2017.11.007.

[34] G. Aceto, G. Bovenzi, D. Ciuonzo, A. Montieri, V. Persico, and A. Pescape, “Characterization and prediction of mobile-app traffic using markov modeling,” *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 907–925, Mar. 2021, doi: 10.1109/TNSM.2021.3051381.

[35] G. Aceto, D. Ciuonzo, A. Montieri, V. Persico, and A. Pescape, “MIRAGE: Mobile-app traffic capture and ground-truth creation,” in *2019 4th International Conference on Computing, Communications and Security (ICCCS)*, Oct. 2019, pp. 1–8, doi: 10.1109/CCCS.2019.8888137.

[36] S. A. Mahmood, K. A. Hussein, Y. N. Jurn, and E. A. Albahrani, “Parallelizable cipher of color image based on two-dimensional chaotic system,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, no. 1, pp. 101–111, Apr. 2020, doi: 10.11591/ijeecs.v18.i1.pp101-111.

[37] Y. Liu, J. Zhang, D. Han, P. Wu, Y. Sun, and Y. S. Moon, “A multidimensional chaotic image encryption algorithm based on the region of interest,” *Multimedia Tools and Applications*, vol. 79, no. 25–26, pp. 17669–17705, Jul. 2020, doi: 10.1007/s11042-020-08645-8.

BIOGRAPHIES OF AUTHORS

Abeer Tariq Maolood received the M.Sc. and Ph.D. in Computer Science from University of Technology, Iraq, 2005 and 2010, respectively. She has around 15 years of teaching experience. Her areas of interests are computer and network security, neural networks, and web applications security. She can be contacted at email: abeer.t.maolood@uotechnology.edu.iq

Ekhlas Khalaf Gbashi earned her Ph.D. in networks security from the Department of computer sciences at the Technology University. Ekhlas earned her bachelor’s and master’s degree in computer sciences from the University of Technology (UOT), Baghdad, Iraq in 1998, 2005. Ekhlas is a faculty member in the computer sciences Department at the University of Technology (UOT) since 2000; where she became a Head of computer security Branch at the UOT from 2016 until 2020. Her research interested focus on in networks security (intrusion detection system), data security, computer networks, comparative education and computer architecture, image processing and artificial intelligence (AI). She can be contacted at email: Ekhlas.K.Gbashi@uotechnology.edu.iq

Eman Shakir Mahmood received the MSc. in Computer Science from University of Technology, Iraq, 2006. She has around 14 years of teaching experience. Her areas of interest's Image processing, operating system, computer security and web applications security. She can be contacted at email: 110036@uotechnology.edu.iq