Chapter 6
Chaos-Based Video Encryption Algorithms

Zhaopin Su¹, Shiguo Lian², Guofu Zhang¹, and Jianguo Jiang¹

¹ School of Computer and Information, Hefei University of Technology, China
² France Telecom R&D Beijing, China

1 Introduction

In recent years, with the development of network technology and multimedia technology, multimedia data, especially video data, are used more and more widely in human society. Some multimedia data applied in entertainment, politics, economics, militaries, industries or education, etc., are necessary to be protected by providing confidentiality, integrity, and ownership or identity. To protect video contents, cryptology, which appears to be an effective way for information security, has been employed in many practical applications[1][2][3]. However, traditional ciphers like DES [4], IDEA [5], RSA[6] and AES[7], are often used for text or binary data, while not suitable for direct video encryption because of the following reasons.

Firstly, as digital videos are usually very large-sized and bulky, encrypting such bulky data with traditional ciphers incurs significant overhead, and it is too expensive for real-time video applications, which require real-time operations, such as displaying, cutting, copying, bit-rate control or recompression.

Secondly, in the case of digital videos, consecutive frames are similar and most likely only few pixels would differ from frame to frame, and such an extremely high data redundancy makes the conventional ciphers fail to obscure all visible information [8].

Additionally, for many practical applications, we would like to have very light encryption that preserves some perceptual information. For example, in a pay-per-view video system [9], a degraded but visible content could potentially influence a consumer to order certain paid services. This is impossible to achieve with traditional ciphers alone, which most likely degrades the data to a perceptually unrecognizable content.

Very recently, an increasing attention has been devoted to the usage of chaotic theory [10][11][12][13][14][15][16] to implement the encryption process. The main advantage of such encryption lies in the observation that a chaotic signal looks like noise for non-authorized users ignoring the mechanism for generating it. Secondly, time evolution of the chaotic signal strongly...
depends on the initial conditions and the control parameters of the generating functions: slight variations in these quantities yield quite different time evolutions. In other words, this means that initial states and control parameters can be efficiently used as keys in an encryption system. What’s more, generating of chaotic signal is often of low cost, which makes it suitable for the encryption of large bulky data [17].

Due to these recognized potential benefits, chaos-based video encryption algorithms are of high interest up to now, and have made great progress. This chapter focuses on chaos-based video encryption algorithms, by reviewing different types of works, investigating the state of the art progress, analyzing the performance evaluation and comparison, and presenting some challenges and open issues.

The organization of this chapter is as follows. In Section 1, backgrounds of chaos-based video encryption algorithms are given. Some requirements of video encryption are described in Section 2. Section 3 introduces the common video coding standards. Section 4 is a comprehensive review on today’s chaos-based video encryption technology of different types. Section 5 and Section 6 compares and discusses the algorithms listed in Section 4, respectively. Some open issues and challenges are presented in Section 7. The last section concludes the chapter.

2 Some Requirements of Video Encryption

Due to some characteristics of digital video, such as large data volumes, high redundancy, interactive operations, and real-time responses, an ideal video encryption algorithm should satisfy some requirements.

1) Security: for video encryption, security is the primary requirement. Generally speaking, an encryption algorithm is regarded as secure if the cost for cracking it is no smaller than the one paid for the authorization of video content. For example, in broadcasting, the news may be of no value after an hour. Thus, if the attacker can not break the encryption algorithm during an hour, then the encryption algorithm may be regarded as secure in this application [18]. Therefore, the usage of chaotic maps in video encryption should guarantee the security of a video.

2) Invariance of compression ratio and constant bit rate: In some applications, it is required that the encryption transformation preserves the size of video data, which is called invariance of compression ratio. An algorithm with invariance of compression ratio can maintain the same storage space or transmission bandwidth, which is called constant bit rate. Sometimes, the encryption stage is allowed to slightly increase the size of a bit stream. In this case, video encryption algorithms should not change compression ratio or at least keep the changes in a small range.

3) Format compliance: In many applications, video data are often encoded or compressed before transmission, which produces the data streams with some
format information. The format information will be used by the decoder to recover the video data successfully. It is desired that the encryption algorithm preserves the multimedia format. In other words, after encrypting the encoded multimedia, ordinary decoders can still decode it without crashing. This property of an encryption algorithm is often called format compliance. When feeding the decoder with the format compliant encrypted data, the produced output seems distorted and randomized [8]. Generally, encrypting the data except the format information will keep the multimedia format. This will support some direct operations (e.g., decoding, playing, bit-rate conversion, etc.) and improve the error robustness in some extent.

4) Transmission error tolerance: Since the real-time transport of video data often occurs in noisy environments, which is especially true in the case of wireless channels [19] [20], the delivered video is prone to bit errors. So, a perfect video encryption algorithm should be insensitive and robust to transmission errors.

5) Demand of real-time: As the demand of real-time of the video transmission and access, encryption and decryption algorithm can not bring much delay to transmission and access. Therefore, the encryption and decryption algorithm must be fast enough to meet the requirements of real-time video applications.

6) Multiple levels of security: A user may be willing to sacrifice some degree of security for the ability to perform more complex video processing. A given cryptosystem provides a certain level of security. Most available cryptographic systems are fully or partially scalable, in the sense that one can choose different security levels. Scalability is usually achieved by allowing variable key sizes or by allowing different number of iterations, or rounds. A higher level of security is achieved with larger key sizes or larger number of rounds [21] [22].

7) Low overhead: Encryption techniques should have minimal overhead in terms of bandwidth requirements and processing power. A low overhead encryption algorithm will provide the end hosts with as much processing power for ensuring high quality video presentation to the user [8]. In addition, providing a secure stream should not significantly increase the bandwidth required to transmit it.

8) Allow degradation: For transmission of secure video streams, the user may be willing to accept a certain degree of degradation in quality in order to achieve a reasonable transport cost for the video session. When degradation is applied to a multimedia content, the content is usually still perceptible to some degree [8]. For instance, in some applications, such as video on demand, database search, etc., it could be desirable to encourage customers to buy the content. For this purpose, one may still see the objects in a degraded video, but the visual quality should be unacceptable for entertainment purposes, so that he prefers to pay to access the full-quality unencrypted content.
3 Some Video File Formats

Due to the huge size of digital videos, they are usually transmitted in compressed formats such as MPEG-x or H.26x. In this section, we will introduce some of them in brief.

3.1 MPEG-x

MPEG (Moving Picture Experts Group) compression standards\cite{23} compress data to form small bits that can be easily transmitted and then decompressed. MPEG achieves its high compression rate by storing only the changes from one frame to another, instead of each entire frame. The video information is then encoded using a technique called Discrete Cosine Transform (DCT). MPEG uses a type of lossy compression, since some data are removed. But the diminishment of data is generally imperceptible to the human eye. MPEG has standardized the following compression standards \cite{23}:

MPEG-1\cite{24}: MPEG-1 is the first MPEG compression standard for audio and video. It was basically designed to allow moving pictures and sound to be encoded into the bit-rate of a Compact Disc. It is used on Video CD, SVCD and can be used for low-quality video on DVD Video. It was used in digital satellite/cable TV services before MPEG-2 became widespread. To meet the low bit requirement, MPEG-1 downsamples the images, as well as uses picture rates of only 24-30 Hz, resulting in a moderate quality.

MPEG-2\cite{25}: MPEG-2 offers resolutions of 720x480 and 1280x720 at 60 fps (frames per second), with full CD-quality audio. This is sufficient for all the major TV standards, including NTSC, and even HDTV. MPEG-2 is used by DVD-ROMs. MPEG-2 can compress a 2 hour video into a few gigabytes. While decompressing an MPEG-2 data stream requires only modest computing power, encoding video in MPEG-2 format requires significantly more processing power.

MPEG-3\cite{26}: Was designed for HDTV but was abandoned in place of using MPEG-2 for HDTV.

MPEG-4\cite{27}: is a graphics and video compression algorithm standard that absorbs many of the features of MPEG-1 and MPEG-2 and other related standards, and uses further coding tools with additional complexity to achieve higher compression factors than MPEG-2. MPEG-4 files are designed to transmit video and images over a narrower bandwidth and can mix video with text, graphics and 2-D and 3-D animation layers.
3.2 H.26x

H.26x family is video compression coding standard in the domain of the ITU-T Video Coding Experts Group (VCEG), and consists of the following standards:

H.261[28]: H.261 was originally designed for transmission over ISDN (Integrated Services Digital Network) lines on which data rates are multiples of 64 kbit/s. The coding algorithm was designed to be able to operate at video bit rates between 40 kbit/s and 2 Mbit/s. The standard supports two video frame sizes: CIF (352x288 luma with 176x144 chroma) and QCIF (176x144 luma with 88x72 chroma) using a 4:2:0 sampling scheme. It also has a backward-compatible trick for sending still picture graphics with 704x576 luma resolution and 352x288 chroma resolution.

H.262[29]: is the second part of the MPEG-2 standard.

H.263[30]: originally designed as a low-bitrate compressed format for videoconferencing and developed as an evolutionary improvement based on experience from H.261, MPEG-1 and MPEG-2 standards.

H.264[31]: is a block-oriented motion-compensation-based codec standard, and contains a number of new features that allow it to compress video much more effectively than older standards and to provide more flexibility for application to a wide variety of network environments. H.264 is used in such applications as players for Blu-ray Discs, videos from YouTube and the iTunes Store, web software such as the Adobe Flash Player and Microsoft Silverlight, broadcast services for DVB and SBTVD, direct-broadcast satellite television services, cable television services, and real-time video conferences.

4 Chaos-Based Video Encryption Algorithms

In the past decade, chaos-based video encryption has been a topic of great interest. According to the relation between compression process and encryption, these proposed algorithms can be classified into two types: encrypting the raw video data, and encrypting the video data in compression process.

4.1 Encrypting the Raw Video Data

This type of chaos-based video encryption algorithm encrypts the raw video data directly with chaotic maps. Among them, some encrypt the raw data completely without considering region-of-interest, [32][33][34][35][36] and some consider the region-of-interest partially or selectively [31][32].

4.1.1 Encryption without Considering Region-of-Interest

Encryption without considering interest regions means to encrypt the video data as binary large objects, pixels, or sets of frames, without taking into con-
consideration video objects or any other kind of regions of semantic information. Thus, it treats the regions fairly without special considerations.

Li et al. [32] proposed a chaotic video encryption scheme (CVES) based on multiple digital chaotic systems, which is independent of any video compression algorithms. In CVES, video data is encrypted frame by frame. First, each plain-block is first XORed by a chaotic signal pseudo-randomly generated from chaotic maps based on perturbation-based algorithm [37], and then substituted by a pseudo-random S-box generated from all chaotic orbits of the chaotic maps. Their detailed analysis have shown that CVES has fair speed and security, and can be realized easily by both hardware and software. Furthermore, CVES can be easily extended to other real-time secure applications.

Ganesan et al. [33] described a public key encryption (PKE) of images and videos based on chebyshev maps, shown as Eq. (1) [38]. In the work, videos in simple terms are considered as a collection of images, and each video is made up of frames and each frame is like a still image. Encrypting video is equal to encrypting each frame by Arnold scrambling [39]. If in a video, the number of frames is too large, the authors propose the use of Phase Scrambling [40] for video encryption instead of Arnold scrambling. The method of Phase Scrambling (See Fig. 1) adds the same random phase structure to the original r, g and b phase structures respectively to randomize the phase of the r, g and b layers of an image. The scrambling operation’s security is not high enough to resist known-plaintext or select-plaintext attacks.

\[
\begin{align*}
T_0(x) &= 1 \\
T_1(x) &= x \\
T_n(x) &= 2 \cdot x \cdot T_{n-1}(x) - T_{n-2}(x), n \geq 2
\end{align*}
\] (1)

Kezia et al. [34] also treated video data as a set of frames, and used a high dimensional Lorenz chaotic system to encrypt each frame by confusing the position of the pixels (hence called LCS). If the frames are large in size, then it is broken into macro-blocks for the encryption. Moreover, their concept of multi-key based on logistic map, whose values cannot be predicted in the long-run, is also employed where each frame is encrypted by a unique key.
instead of changing the key for a particular number of frames. The position confusion operation is weak when it is used alone for data encryption.

Mao et al. [35] and Lian et al. [36] first extended the standard two-dimensional baker map to a three-dimensional setting, then constructed a fast and secure encryption scheme (FSES) (see Fig. 2). In their scheme, to make known-plaintext attack infeasible, an XOR plus modulo (mod) operation (see Eq. (2)) is inserted between every two adjacent rounds of chaotic map based confusion. This kind of scheme combines confusion and diffusion, and aims to obey traditional block cipher’s principles.

\[ C(k) = \phi(k) \oplus \{ [I(k) + \phi(k)] \mod N \} \oplus C(k - 1) \quad (2) \]

The chaos-based video encryption algorithms mentioned above deal with video data as binary large objects, pixels, or sets of frames, without taking into account of regions-of-interest. These regions may need better protection or can be the only regions that need protection, depending on the practical applications.

4.1.2 The Encryption Considering Regions-of-Interest

Generally speaking, for the video data, a region of interest means human video objects or any other kind of regions of semantic information. In many practical applications, it is not necessary or suitable to encrypt all video data, while just regions of interest. In this issue, researchers have proposed some encryption algorithms.

Tzouveli et al. [41] proposed a human video object encryption system (HVOE) based on the chaotic logistic map (see Fig. 3). In their system, face regions are first efficiently detected, and afterwards body regions are extracted using geometric information of the location of face regions. Then, the pixels of extracted human video objects are encrypted using an iterative cipher module, which is based on logistic map and a feedback mechanism responsible for mixing the current encryption parameters with encrypted information of the previous step. The encryption of each plain pixel depends on the key, the value of the previous cipher pixel and the output of the logistic map. This method can save a great amount of computational resources and time devoted for encrypting the whole contents of a video file. This method’s
security depends on the chaotic sequence’s randomness and the detection of face region.

Ntalianis et al. [42] proposed a video object based chaotic encryption system (VOCE) (Fig. 4). Initially, stereoscopic pairs are analyzed and video objects are automatically extracted based on the appropriate fusion of color information according to depth constraints. Next, for each video object, multiresolution decomposition is performed and the pixels of the lowest resolution level are encrypted using a chaotic cipher module combining a simple chaotic stream cipher and two simple chaotic block ciphers (with time variant S-boxes) to implement a complex product cipher. Finally, the encrypted regions are propagated to the higher resolution levels and the encryption process is repeated until the highest level is reached. The system presents robustness against known cryptanalytic attacks, enables layered access of multimedia content and the overall security can be enhanced due to region topology. This scheme’s security depends on the encryption algorithm and the video object detection.

Encrypting the raw video data before encoding or compression process removes a lot of redundancy, which results in a very poor compression ratio. Additionally, it changes video data format, so that the encrypted video cannot
be displayed without decrypting it firstly. So, some researchers have proposed that the encryption should be implemented in compression process.

4.2 Encrypting the Video Data in Compression Process

Encrypting the video data in compression process means realizing encryption in the encoding process before entropy coding, i.e. CAVLC(Context-adaptive variable-length coding), CABAC(Context-adaptive binary arithmetic coding), VLC(variable length coding), RLC(run length coding), Golomb, Huffman, etc. Till now, some algorithms have been proposed for MPEG, and some for H.26x.

4.2.1 Encryption for MPEG

For MPEG, the representative works are done by Yang et al. [43], Lian et al. [44][45] and Hamdi et al. [46]. Yang et al. [43] used double coupling logistic maps (shown as Eq. (3)) to scramble the DCT coefficients of every I-frame of the video, and then used another chaotic map (shown as Eq. (4)) to encrypt the DCT coefficients of the scrambled I-frame (hence called DCLM). The process of encryption is shown in Fig. 5. In the whole process, five keys are introduced, and the key space is large, which makes brute-force attack difficult. Moreover, as I-frames do not refer to any other frame and are the beginning of decoding, their changes can greatly influence the B-frames and P-frames, which makes the DCLM effective in video protection. Besides, DCLM, only encrypting the DCT coefficients of I-frames, brings little consumed time, and is feasible for real-time applications. However, considering that there are some macro blocks in B-frame or P-frame, which are encoded without referring to I-frame, these blocks will be left unencrypted. Thus, some video contents may be intelligible, and thus the encryption scheme is not secure enough.

\[
\begin{align*}
x_{n+1} &= \mu x_n (1 - x_n) \\
y_{n+1} &= \mu y_n (1 - y_n) \\
x(i+1) &= 1 - \mu x^2(i)
\end{align*}
\]

(3) (4)

Lian et al. [44] constructed an efficient image/video encryption scheme (EES) based on 2D coupled map lattice (CML) [47], which is a kind of spatiotemporal chaos. The chaotic lattices are used to generate pseudo-random sequences and then encrypt some sensitive parameters during the video compression process. For example, for MPEG2 videos, only the intra-blocks in each frame are encrypted. Fig. 6 gives the architecture of the encryption scheme. In compression, after pre-encoding (i.e., color space transformation),
block partitioning (each block is in 88 size), DCT transformation and quantization, the blocks are encrypted by the proposed cipher one by one, and the cipher-blocks are then post-encoded (i.e., zig-zag scan and VLC). The proposed scheme satisfies the requirement of secure encryption principles, the encrypted videos are secure in perception, the encryption operation does not change the compression ratio apparently, and increases little computational cost compared with video compression. The scheme’s cryptographic security depends on the randomness of the chaotic sequences generated by 2D coupled map lattice.

Lian et al. [45] presented a video encryption algorithm which combines encryption process with MPEG-2 encoding process (hence called VEM2). In the algorithm, I-frame, B-frame and P-frame of MPEG-2 video are encrypted with different methods, respectively. That is, for I-frames, the macroblocks are inter-permuted by color-plane confusion. For each intra-macroblock, the DCT coefficients are permuted by coefficient confusion. The DCT coefficients’ signs of each intra-macroblock and motion vectors of each inter-macroblock are modulated by chaotic sequence generated by the chaotic sequence generator proposed in [48]. All these encryption processes are controlled by a key generation and distribution system based on Logistic map [49]. The analysis and experiments have shown that the VEM2 is secure against brute-force attack and known-plaintext attack. Moreover, the VEM2 is of low computational complexity, costs little time, supports direct bit-rate control and is more robust to transmission errors.
In addition, Hamdi et al. [46] also proposed a progressive Chaotic Video Encryption Scheme (PCVE) for MPEG-4 coding. In their scheme, multi-dimensional chaotic maps [50] are used to build chaotic multi-resolution transforms, which can introduce randomness in the selection of wavelet filters and de-correlate the video data. The major advantage of PCVE is that a client can access to multiple resolutions of the streamed video. These resolutions vary according to the security level of the client as well as the networking and processing capabilities. Moreover, the PCVE may be secure against statistical attacks and known-plaintext cryptanalysis. Additionally, it is of low computational complexity, and does not affect the compression performance ensured by the MPEG-4 coder. Besides, The bit rate control functionality makes it compatible with video transmission in wireless networks.

4.2.2 Encryption for H.26x

For H.26x, the representative works are done by Jian et al [51] and Chiaraluce et al. [52].

For H.263, Jian et al [51] proposed chaos-based encryption algorithm (CBEA) for the H.263 video-conference coding standard. A sawtooth-like chaotic map is first used to generate a pseudo-random bit sequence (PRBS). Then, according to the PRBS, all of the DC coefficients, part of the AC coefficients of I blocks as well as Motion Vectors (MVs) are encrypted in the video coding. The full encryption algorithm is shown in Fig. 7, where the cipher operations have been seamlessly integrated into the H.263 encoding process, i.e., before RLC and packaging. Although the encryption processing doesn’t interfere the motion features of the videos, the algorithm may introduce slight computational overhead and slight data inflation in video encoding. The scheme’s security depends on the chaotic sequence’s randomness.

Chiaraluce et al. [52] presented a selective encryption algorithm (SEA) for the H.263+ videos, which employs suitably arranged three different chaotic functions (see Fig. 8) to encrypt the video data selectively. In Fig. 8, video data include the most significant bit in the DC coefficients of DCT (Discrete Cosine Transform), the AC coefficients of I-MB’s (Intra MacroBlocks), the sign bit of the AC coefficients of the PMB’s (Predicted MacroBlocks) and the

![Fig. 7 Chaos-based encryption algorithm in [51]](image-url)
sign bit of the Motion Vectors. And $CM_1$ is the skew tent map \cite{53}, $CM_2$ is a saw-tooth likewise map \cite{54}, and $CM_3$ is the logistic map \cite{49}. The outputs of the $CM_1$ and $CM_2$ are added, and then the addition is scaled to be an integer between 0 and 255. Each scaled integer is used as the initial condition of the third map to generate a 64-size key stream to mask the plaintext with XOR operation. In order to increase the security level against known/chosen-plaintext attacks, it was suggested to change the key every 30 frames. The algorithm introduces a modest delay, offers good security and the ability to reconstruct perfectly the image, and gets a good compromise between the need to improve security while maintaining a limited additional processing time. These properties make it suitable for video applications that have real time or almost real time requirements.

4.3 Encrypting the Compressed Video Data

Encrypting the compressed video data means realizing encryption after entropy-encoding and before package. The representative works are done by Lian et al. \cite{55} \cite{56} and Qian et al. \cite{57}.

Lian et al. \cite{55} constructed a chaotic stream cipher with random feedback mode based on a discrete piecewise linear chaotic map \cite{58}, and then encrypted both the intra-macroblocks (all the macroblocks in I-frame and some intra-encoded macroblocks in P/B-frame) and the motion vectors’ signs segment by segment (hence called CSCF), which is shown in Fig. 9. The whole encryption process is achieved after VLC and before packaging. The encryption scheme is of high key sensitivity, secure in perception, format compliant, and error robust. Besides, the encryption/decryption process does not affect the compression/decompression process greatly. The cryptographic security depends on the chaotic sequence’s randomness.

In \cite{56}, a fast video encryption scheme is proposed combining with MPEG-4 codec (hence called VEM4). In the scheme, the file format information, such as file header, packet header, and so on, are left unencrypted in order to support such operation as bit-rate control; the motion vectors, subbands, code blocks or bit-planes are partially encrypted by a stream cipher based on a
modified chaotic neural network. Moreover, for each encoding-pass, the chaotic binary sequence is generated from different initial-condition based on logistic map. Thus, if one encoding-pass cannot be synchronized because of transmission errors, the other ones can still be decrypted correctly. The encryption scheme is of high security in perception, of low computation complexity, and secure against brute-force attack, statistic attack or differential attack. And it keeps compression ratio and file format unchanged, supports direct bit-rate control, and keeps the error-robustness unchanged. The scheme’s cryptographic security depends on the chaotic sequence’s randomness.

Qian et al. proposed a multiple chaotic system (MCS) for MPEG-2 which combines the partial encryption with block permutation and confusion. In their system, three chaotic or hyperchaotic maps, namely Logistics Map, 2-D Baker Map and 4-D hyperchaotic Map, are introduced for stream partial encryptions, block permutation, confusion after block permutation, respectively. Moreover, stream ciphers encrypt only DC coefficients by XOR operation after DCT and quantization when compressing the video data, and block permutation and confusion are carried out after the video compression, respectively (See Fig. 10). The algorithm is secure, efficient, and of low computational complexity. Besides, it nearly brings no data expansion.

Some performances are often considered to evaluate a video encryption scheme, e.g., security analysis, encryption speed, compression ratio and error robustness.

5 Performance Evaluation

Some performances are often considered to evaluate a video encryption scheme, e.g., security analysis, encryption speed, compression ratio and error robustness.
5.1 Security Analysis

Security of an algorithm is generally evaluated by the perceptual experiments, key space analysis, key sensitivity analysis, and the ability against attacks.

The perceptual experimental result is achieved by a group of comparison between frames of the original video and those of the encrypted one. Besides, some works decrypt the encrypted video to examine the effects of their encryption.

Key space of an encryption algorithm is generally defined as the number of encryption/decryption key pairs that are available in the cryptosystem. Assume $k_i$ denotes a key and $K$ represents a finite set of possible keys, the key space can be expressed as $K = \{k_1, k_2, \ldots, k_r\}$, where $r$ is the number of key. To make brute-force attack infeasible, the size of key space should be large enough. For chaos-based encryptions, the chaotic sequence generator should produce chaotic ciphers with good randomness, which can be tested by long period, large linear complexity, randomness and proper order of correlation immunity [67].

Key sensitivity of a chaotic cipher refers to the initial states sensitivity and control parameters sensitivity of chaotic map. A typical key sensitivity test is performed according to the following steps: First, assume a frame of a video is encrypted by using the key "$K_1=0123456789$". Then, the same frame is encrypted by using the key "$K_2=1123456789$", which changes the least significant bit of $K_1$. Finally, the above two encrypted frames, encrypted by $K_1$ and $K_2$ respectively, are compared, and cross-correlation curve between the two encrypted frames is analyzed.

A good cipher can avoid potential attacks. In general, brute-force attack is analyzed by key space analysis. Known-plaintext attack and chosen-plaintext attack can be tested by comparing the original frame of a video and the decrypted one. Differential attack test can be achieved through measuring the percentage $p$ of different pixel numbers (see Eq. (5) and Eq. (6)) between two encrypted images, $I_1$ and $I_2$ (the width and height is $W$ and $H$, respectively), whose corresponding plain-images have only one pixel’s difference.

$$p = \frac{\sum_{i,j} D(i,j)}{W \cdot H} \cdot 100\%, i = 0, 1, \ldots, W - 1, j = 0, 1, \ldots, H - 1$$  \hspace{1cm} (5)

$$D(i, j) = \begin{cases} 0, & I_1(i,j) = I_2(i,j) \\ 1, & \text{otherwise} \end{cases}$$  \hspace{1cm} (6)

5.2 Encryption Speed Test

The encryption time is tested in three manners: absolute encryption time, relative encryption time ratio, and computation complexity analysis. Absolute encryption time refers to the assumed time for encrypting a video, and
its measuring unit is second. Relative encryption time ratio refers to the time ratio between encryption and compression. Computation complexity of an encryption scheme depends on the cost of the chaos-based cipher and the video data volumes to be encrypted.

If the computational cost or assumed time of a video encryption scheme is very little compared with video compression, it is considered to be suitable for real-time applications.

5.3 Compression Ratio Test

In general, the compression ratio is tested by comparing the original compressed data volumes and encrypted and compressed data volumes. Considering that the compression coder often produces the data stream with a given bit-rate, the compression ratio test may be measured by the video quality under certain bit rate. The common measurement of video quality is $\text{PSNR}$ (Peak Signal-to-Noise Ratio) shown as Eq. (7) and Eq. (8), where $B$ is sampling frequency, $I$ and $I'$ represent an original $mn$ frame and the encrypted one, respectively.

$$\text{PSNR} = 10 \cdot \log_{10} \left( \frac{(2^B - 1)^2}{MSE} \right)$$  \hspace{1cm} (7)

$$MSE = \frac{1}{m \cdot n} \cdot \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - I'(i, j)]^2$$  \hspace{1cm} (8)

5.4 Error-Robustness Test

If an encryption scheme does not change file format, and a slight change in one pixel does not spread to others, it is of lower sensitivity to transmission errors.

The general test method for error-robustness is analyzing the relationship (usually expressed by a curve) between the quality $\text{PSNR}$ of the decrypted frames and the number of bit-error happened in the encrypted frames. Besides, error-robustness can be tested through correct decryption of an encrypted video, even if a frame is corrupted or lost in its transmission.

6 Performance Comparison

In this section, we compare the performance of different encryption algorithms mentioned above. Here, various aspects listed in Section 2 are considered, and contrast results are shown in Table 1.
Table 1 Comparison of chaos-based video encryption algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Security</th>
<th>ICR</th>
<th>FC</th>
<th>TET</th>
<th>Real-Time</th>
<th>MLS</th>
<th>LO</th>
<th>AD</th>
</tr>
</thead>
<tbody>
<tr>
<td>CVES[32]</td>
<td>H</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>PKE[33]</td>
<td>L</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>LCS[34]</td>
<td>L</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>FSES[35]</td>
<td>H</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>HVOE[41]</td>
<td>L</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>VOCE[42]</td>
<td>M</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>DCLM[43]</td>
<td>L</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>EES[44]</td>
<td>M</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>VEM2[45]</td>
<td>M</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>PCVE[46]</td>
<td>M</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>CBEA[51]</td>
<td>M</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>SEA[52]</td>
<td>M</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>CSCF[53]</td>
<td>M</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>VEM4[54]</td>
<td>M</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>MCS[57]</td>
<td>H</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

ICR: Invariance of compression ratio; FC: Format compliance; TET: Transmission error tolerance; MLS: Multiple levels of security; LO: Low overhead; AD: Allow degradation; L: Low security; M: Middle security; H: High security.

From Table 1, we get the following conclusions:

1) The encryption algorithms, CVES, PKE, LCS and FSES, encrypt the video data completely, without considering interest regions. Their security depends on the proposed chaotic ciphers. Generally, if the ciphers are well-designed, they are often of higher security, higher complexity, and higher overhead than other types. So, they are more suitable for secure video storing than for real-time transmission.

2) The encryption algorithms considering interest regions (HVOE, VOCE) encrypt only the interest regions, and leave the rest (such as background) unprotected. They are of lower computation complexity and lower overhead, and more suitable for real-time applications than that without considering interest regions. Their cryptographic security depends on the adopted chaotic cipher and the region selection. These algorithms that encrypt the raw video data directly cannot preserve invariance of compression ratio, and change the video format, so that the encrypted video cannot be displayed without decrypting it firstly. Besides, they do not consider the compression process, especially lossy compression, which may result in degradation after video decoding, and bring difficulty to decryption.

3) The algorithms that encrypt the video data in compression process belong to partial or selective encryption, and are often of lower complexity than those encrypt the raw video data directly. However, some of them (DCLM, SEA)
change the compression ratio for they change the statistical characteristics of DCT coefficients. Interestingly, some of them can keep file format unchanged. Thus, these algorithms support direct bit rate control, that is, they permit to re-compress the encoded and encrypted video before decrypting it firstly, and save much time for secure transcoding. Therefore, they are more suitable for real-time applications, such as wireless multimedia network or multimedia transmission over narrow bands.

4) The algorithms that encrypt the compressed video data can not only preserve invariance of compression ratio and format compliance, but also be of low overhead. Additionally, it is of low-cost and is easy to be realized. For these advantages, it is suitable for real-time required applications, such as video transmission or video access. However, as the video stream after entropy encoding may have a certain structure or syntax, the encryption scheme may destroy the structure of the video stream. And thus, these algorithms, without considering the rules of package before transmission, may bring error spreading when the transmission error happens.

5) For the algorithms in Table 1, many of them don’t give any illustration whether their algorithms are insensitive and robust to transmission errors except LCS, VEM2, CSCF and VEM4.

7 Discussions

From the above survey on the issues of chaos-based video encryption, we can learn many valuable experiences on how to design a video encryption scheme based on chaotic maps. However, there are still some questions to be discussed:

Now, the security of different chaos-based encryption algorithms is evaluated by different means, including the perceptual results, key space, key sensitivity, ability against attacks, and so on. How to assess a chaos-based cipher and also the video specific encryption algorithm in a general manner is still an open issue.

Most of existing chaos-based video encryption schemes provide different levels of video security. Various practical applications may demand different encryption approaches with certain security levels. Consequently, how to define the security levels in detail and how to choose the appropriate security level for certain application will be interesting topics.

Encrypting more video data results in higher level of security, meanwhile, it also produces more computational complexity and costs much more time. Therefore, how to get a good tradeoff between security and time-efficiency is another important issue.

As is known that, encryption algorithms are often sensitive to transmission errors, for a slight change in cipher text often causes great changes in the decoded data. The existing chaos-based video encryption schemes lack of error-robustness test in practical transmission environments. Thus, how to
design a chaos-based video encryption scheme with strong error-robustness is also an challenging task.

8 Conclusions

Video encryption plays a more and more important role in today’s multimedia world. And chaos theory provides a fast and practical solution for the design of digital cipher for video encryption. Many efforts have been devoted to study the security issue, and some valuable algorithms can be used as the fundamentals of future research.

Although chaos-based video encryption appears to be promising, it is not yet mature. More efforts are needed for its further development toward practical applications with high security, invariance of compression ratio, format compliance, strong transmission error tolerance, real-time, multiple levels of security, low overhead and allow degradation.

References

2. Lian, S., Zhang, Y.: Handbook of research on secure multimedia distribution. IGI Global (formerly Idea Group, Inc.), Hershey, Pennsylvania (2009)
44. Lian, S.: Efficient image or video encryption based on spatiotemporal chaos system. Chaos, Solitons and Fractals 40, 2509–2519 (2009)